Lotus software

IBM

Security Considerations in Lotus Notes and Domino 7:

Making Great Security Easier to Implement

- New security features in Lotus Notes and Domino 7
- Best practices for implementing Lotus Notes and Domino security
- Applicable scenarios and examples

Frederic Dahm
Paul Ryan
Richard Schwartz
Amy Smith
Dieter Stalder

Redbooks

ibm.com/redbooks

International Technical Support Organization

Security Considerations in Lotus Notes and Domino 7: Making Great Security Easier to Implement

March 2006

SG24-7256-00

Note: Before using this information and the product it supports, read the information in “Notices” on page vii.

First Edition (March 2006)

This edition applies to IBM Lotus Notes and Domino Release 7, with some applicable references to IBM Lotus Notes and Domino Releases 5 and 6.

Contents

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX 5L™	iNotes™	Redbooks (logo) ™
AIX®	iSeries™	Redbooks™
Domino®	Lotus Notes®	Sametime®
@server®	Lotus®	Tivoli®
i5/OS®	Lotusphere®	WebSphere®
ibm.com®	Notes®	Workplace™
IBM®	QuickPlace®	zSeries®

The following terms are trademarks of other companies:

Java, JavaScript, Solaris, Sun, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

ActiveX, Microsoft, MS-DOS, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Preface

This IBM Redbook is the fourth in a series about IBM Lotus® security to be published. The previous IBM Redbooks™ about the topic are, in chronological order, *The Domino Defense: Security in Lotus Notes 4.5 and the Internet*, SG24-4848 (for Release 4.5 of IBM Lotus Notes® and Domino®), *Lotus Notes and Domino R5.0 Security Infrastructure Revealed*, SG24-5341 (for Release 5.0 of Lotus Notes and Domino), and *Lotus Security Handbook*, SG24-7017 (for Release 6.0 of Lotus Notes and Domino, including extended products).

The primary goal of these publications was to focus on the strong security that has always been part of the family of Lotus products. The present publication continues down the path set by these previous Redbooks, offering, as with each previous release, information about key features and functionalities pertaining to the security aspects of Lotus Notes and Domino Release 7.0.x, as well as best practices to implement these new features and functionalities.

The security enhancements in Notes and Domino 7 build on the security features in Releases 6.0.x and 6.5.x. This is reflected in the size of this publication, because we assume that you are familiar with the concepts and information written in the *Lotus Security Handbook* prior to reading this publication. For the sake of brevity, little will be repeated in this book prior to covering the new security features and functionality.

Nonetheless, despite its size and the scope of the topics that we cover, this publication gives technical people the necessary details for understanding and correctly implementing the new security features present in Release 7.0.x of Lotus Notes, Domino, and the extended products. This book also helps IT managers understand the new security features and understand how these services can be provided to the end-user population in an effort to further enhance the range of security services and functionalities to meet the business needs of the organization, securely and effectively.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Cambridge Center.

Frederic Dahm is a Systems Architect for IBM Software Services for Lotus based in Montreal, Canada (and was, until recently, living and working in Switzerland). He brings to the team 15 years of experience, including installing and working with Lotus Notes 1.0. In addition, he has spoken at a number of conferences and has provided his help as a Lotus security subject matter expert in a number of engagements worldwide.

Paul Ryan, Principal and Architect of Process Stream Technologies in Denver, Colorado, delivers application architecture and development services, including unique integrations of outside technologies with the IBM Lotus Notes/Domino platform. Such integrations are often done using the efficiencies of the low-level Notes C API. For a prominent example, Paul delivered for PGP Corporation a deep integration of its e-mail security tooling with Notes/Domino (http://www.pgp.com), including smartcard integration. While a small consultancy itself, Process Stream adds heft by tapping the Penumbra Group, a consortium of select IBM Business Partners who collaborate to deliver joint, best-of-breed solutions and products (http://www.PenumbraGroup.org).

Richard Schwartz has been working in the messaging and collaboration industry since 1985. He is the founder of RHS Consulting (http://www.rhs.com), a New Hampshire-based IBM Business Partner, and also a founding member of the Penumbra Group. He has spoken on a wide variety of topics at industry conferences, including Lotusphere® and he has written numerous articles about development, administration and security topics for Notes- and Domino-related magazines and Web sites. He contributed several chapters to *Lotus Notes and Domino 6 Programming Bible*, and this is the second Redbook on which he has worked.

Amy Smith is a Principal Information Developer on the IBM User Experience Information Development team and is based in Westford, MA. She writes primarily about Domino and IBM Workplace™ security. She has also written a number of articles for the Lotus Developer Domain and authored or coauthored several white papers. Amy has more than 20 years of experience in technical writing in the computer and financial services industries and holds a masters' degree in technical and professional writing from Northeastern University.

Dieter Stalder founded STDI Consulting Inc. in 1994 (http://www.stdi.com). Dieter's interest in all aspects of e-mail security started early in 2002 when the spam problem was still in its infancy. Since then, Dieter monitors developments in spam mail to find new ways to filter and manage e-mail. All of his research goes into spamJam, an e-mail management application distributed by Granite Software. Another important part of e-mail security is x.509 signed and encrypted e-mail. This standard provides powerful possibilities, but unfortunately, has only been slowly accepted.

John Bergland is a project leader at the International Technical Support Organization, Cambridge Center. He manages projects that produce Redbooks about Lotus software products. Before joining the ITSO in 2003, John worked as an Advisory IT Specialist with IBM Software Services for Lotus, specializing in Notes and Domino messaging and collaborative solutions.

Additional contributors

Thanks to the following people for their contributions to this project:

- Kevin Lynch, Domino Directory/Domino Security, Lotus Software, IBM, Westford, MA
- Katherine Emling, Domino Security/Domino Platform Strategy, Lotus Software, IBM, Westford, MA
- David Kern, Domino Security, Lotus Software, IBM, Westford, MA
- Chuck Connell, President, CHC-3, http://www.chc-3.com
- George Chiesa, DotNSF, http://www.dotnsf.com/
- Markus Seitz, Head of Research & Development, ICODEX Software A.G., Eisenstadt, Austria, http://www.icodex.com

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our Redbooks to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  ibm.com/redbooks

- Send your comments in an e-mail to:

  redbook@us.ibm.com

- Mail your comments to:

  IBM® Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

1

Introduction to security enhancements for Lotus Notes and Domino 7

Strong security has always been part of the family of Lotus software products. More notably, it has been a feature that has made Lotus Notes and Domino an industry leader for security-rich messaging, calendar, and scheduling capabilities. with a robust platform for collaborative applications. With Lotus Notes and Domino 7, IBM extends the reach of Lotus Domino messaging and collaboration solutions while continuing to leverage your IT and application investments. The new version offers capabilities to support more people with fewer servers, to simplify administration, and to provide tighter integration with Web standards.

In this IBM Redbook, we discuss specific security and anti-spam enhancements that have been incorporated into Notes and Domino 7.0.x. The topics include:

- Custom password policies
- Support for larger keys in Notes and Domino 7
- Smartcards
- Domino Web Access
- Customizing password and certificate expiration
- Public key checking enhancements
- ID recovery enhancements
- Single sign-on (SSO) and name mapping in Domino

The goal of this chapter is to offer a quick overview of these enhancements as a means to help understand their value in improving the security of your IT infrastructure.

1.1 Building on a solid foundation

“A skyscraper doesn't start at street level. In fact, the taller the building, the deeper the foundation.”

Author Unknown

Most people have heard this phrase many times, so much so that it has become a cliché. It is often used to express the idea that great things do not get built overnight and do not get built without any planning, or foundations so to speak.

This is true of IBM Lotus Notes and Domino, and at first glance, we could justifiably leave it at that because it is a correct statement to make. After all, the groupware tool has been available for more than a decade and a half and has never succumbed to a major attack and has never shown any major vulnerability of any kind making the tool easy to attack, given a proper setting of the security functionality and the application of best practices, which have been often shared by Lotus and IBM through a number of publications, such as the previous Redbooks in this Lotus security series.

Lotus Notes and Domino have shown time and time again that the foundations on which their security model was built is deep, solid, well designed, and well built. The Notes client and Domino server security functionalities are designed with simplicity in mind, so as to be easy to plan, design, deploy, and use. At the same time, these security functionalities are built on excellent security standards (RFCs, PKCS, and so on) and excellent security toolkits from RSA Security.

But if we left it at comparing the security of Lotus Notes and Domino to a simple cliché, we would not be giving the whole story and we would not be fair to the capabilities of Notes, Domino, and the extended products that now integrate with them. The real strengths of Lotus Notes and Domino in matters of security lay beyond the simple concept of deep foundations.

First, for a building to be tall, foundations not only have to be deep, but they also have to be strong. It stands to reason that the taller the building, the heavier it is and the stronger the foundations have to be. The security model of Lotus Notes and Domino is strong because it is granular and diversified in nature. You do not have encryption only, but a number of mechanisms and functionalities to ensure the authenticity of the user, of the operation to be carried out, and of the code that carries out the security operation needed to keep the environment safe and absent from security risks.

Second, those foundations need to be strong, yet must be able to react to unexpected events and circumstances. Deep and strong foundations that are overly rigid have proven over time remarkably inefficient when earthquakes strike. There have been some very well-designed structures and buildings that were thrown out of plumb, resulting in substantial damage to them with partial collapse in some cases. In some cases, theses buildings were also shifted off their foundations. The buildings that fared best were those whose foundations were able to offer a dynamic response to the earthquake's ground motion. So, not only depth and strength is required, but also dynamic response.

This is also a characteristic that Notes and Domino possess. With the constant evolution of computing technologies making more and more information available to users in a ever faster and more efficient manner to them, it has been necessary for Lotus Notes and Domino to evolve with them. In addition, with new malicious software (malware) appearing at an ever-increasing pace, it has been also necessary for Notes and Domino to evolve so as to ensure that the response to potential threats can be dynamic in nature and ensure little or no damage nor compromise to the information contained within the Notes and Domino infrastructure.

But even if we left it at comparing the security of Lotus Notes and Domino to this improved cliché, we would still not be providing the whole story.

This is because the cliché works really well for tall buildings that never change (for example, Empire State Building in New York, Place Ville-Marie in Montreal, the Sears Tower in Chicago, la Tour Montmartre in Paris, and the Swiss Re Gherkin in London). However, these buildings represent but a small fraction of the all the buildings in the world, and most buildings tend to evolve over time: Annexes and floors are added, floors are rearranged within the infrastructure, and so on. All this occurs to accommodate the needs of the occupants of the building, forcing the foundations to change and adapt to these new needs and demands placed on them.

This is also what has been asked over the years of Lotus Notes and Domino. The groupware tool was designed, created, and made available before most security standards existed. As new standards emerged and a new security technologies were used and embraced by people and organizations worldwide, Lotus evolved the security of the groupware infrastructure to incorporate these standards efficiently and effectively, ever extending the security model without introducing any real vulnerabilities. Lotus Notes and Domino offer more services, features, and functionalities than ever before, and all along, the security mechanisms help ensure that all these new features can be used securely by the users and administrators.

So, if someone tells a story about tall building and deep foundations, you will know that there is more to it than that.

This is, in essence, the purpose and mission of this book. We hope that you take the time to read and benefit from the knowledge imparted within it, because we believe that Notes and Domino encompass the very best in matters of security. They are the shining towers that everyone else would like to have built, that have evolved over time, and have proven time and again their resilience at anything that came their way.

1.2 The new security features

In this book, we discuss specific security and anti-spam enhancements that have been incorporated into Notes and Domino Release 7.0.x. While we discuss these in greater detail in each of their respective chapters, we take a moment to provide an overview here.

1.2.1 Custom password policies

The ability to implement password restrictions on a policy basis has been added to Lotus Domino. This new feature enables administrators to enforce password requirements that will fit almost any set of corporate or government security requirements.

Custom password policies are created and applied through a security policy settings document. Through a custom password policy, administrators can restrict or prohibit the use of the following items in user passwords:

- User name as part of the password
- Repeating characters
- Unique characters
- Use of special characters (such as punctuation), numbers, and uppercase and lowercase characters
- Starting or ending passwords with certain character types
- Combinations of mixed-case characters and non-alphabetic characters

This represents a big improvement to the previous incarnations of enforcement mechanisms for passwords.

1.2.2 Support for larger keys in Notes and Domino 7

As computing power increases, so does the need to extend key lengths to protect against brute force attacks.

In Release 6.0, Notes and Domino can use 1024-bit RSA keys, but cannot not generate them, and can use 128-bit RC4 keys, but cannot use 128-bit RC2 keys. With the advent of 6.0.4 and 6.5.1, Notes and Domino continued to use 1024-bit RSA but can now use 128-bit RC2 keys (however, Notes and Domino cannot generate these RC2 keys).

In Release 7.0, enhancements in Notes and Domino permit 1024-bit RSA keys to be used and generated. In addition, 128-bit RC2 keys can also be used and generated, and there is underlying support for 2048-bit RSA keys.

To help with implementing larger keys, we use key rollover, the process used to update the set of Notes public and private keys that is stored in user and server ID files. Use this to periodically replace this set of keys as a precaution against undetected compromise of the private key, as a remedy to recover from a known compromise of the private key, to increase security by updating to a larger key.

1.2.3 Smartcards

Smartcards were introduced with Lotus Notes Release 6.0, permitting Notes users to use a smartcard with their user ID to log in to Notes. Smartcard use continues to require the installation of a smartcard reader on the user's computer, along with the smartcard software and drivers. The advantage, and one of the reasons for using a smartcard with Notes, is that the smartcard locks the user ID. Logging in to Notes with a smartcard requires the smartcard, the user ID, and the user's smartcard PIN.

Notes smartcard functionality has been extended in Release 7.0 to the following three areas:

- To secure the Notes ID file itself
- To use a private X.509 key stored on the smartcard for Secure/Multipurpose Internet Mail Extensions (S/MIME) private key, data security operations (signing and decryption) and in Secure Sockets Layer (SSL) client authentication
- With large key Notes IDs (1024+ bit), to use the private Notes key in client authentication and in private key, data security operations (signing and decryption)

1.2.4 Domino Web Access

Lotus Domino Web Access Release 7.0 introduces a couple of new security features: one that addresses a security concern stemming from the use of the Web browser as the client used for Domino Web Access, the other that brings parity between Domino Web Access and Lotus Notes in the area of secure Internet messaging.

The first, Browser Cache Management, improves client performance and security of Domino Web Access sessions on Microsoft® Internet Explorer, as it controls which entries are stored in the cache and which are removed when the Domino Web Access session ends.

The second, S/MIME support, permits the exchange of secure messages using a Web browser in conjunction with Domino Web Access.

1.2.5 Customizing password and certificate expiration

New functionality now exists in Release 7 that permits the customization of password and certificate expiration. In the Security Settings form, there have been specific additions with respect to this new functionality.

On the Password Management tab, on the Password Management Basics subtab, there is the Waiting Period field that permits you to specify the number of days prior to password expiration at which the user receives an expiration warning message. Tied to that is the Custom Warning Message field, which is a custom warning message for Notes client users whose passwords have passed the expiration threshold specified in the Warning Period field.

The Keys and Certificates tab also provides Warning Period and Custom Warning Message fields. The Warning Period field here again specifies the number of days prior to password expiration at which the user receives an expiration warning message. Tied to that field is also the Custom Warning Message field, which is a custom warning message to users whose certificates have passed the expiration threshold.

1.2.6 Public key checking enhancements

The signatures on user and server certificates exchanged during authentication are always checked. It is possible in Release 7.0 to enable an additional level of verification for public keys, by having the value of the key passed in the certificates checked against the value of the key listed in the Domino Directory. It is possible for users to authenticate with a server, but have a mismatch between the value of the public keys in their certificates and what is listed for them in the Domino Directory.

This extra level of key verification protects against misuse of a lost or compromised ID file. Typically, if an ID file is lost, its owner needs to be registered to create a new ID file and directory entry; and if the ID file has been compromised, the owner's public and private keys need to be rolled over and that new set of keys needs to be certified (thus updating the directory entry). By enabling directory-level key checking, an attacker in possession of the old ID file will not be able to use it to access the server, even though that old ID file might contain a valid certificate.

In Release 7, you can also control whether a log message is generated if authentication succeeds but a mismatch is detected. This enables administrators to detect when the ID file contents have gotten out of sync with directory entries, but to do so without preventing those users from authenticating because of public key mismatches.

1.2.7 ID recovery enhancements

For each administrator, the user's ID file contains a recovery password that is randomly generated and encrypted with the administrator's public key. The password is unique for each administrator and user. For example, the administrator has a unique recovery password for a specific user and that password is stored in the user's ID file.

In Domino Release 7.0, it is now possible to select the number of characters, or password length, for recovery passwords, which helps determine password strength, or likelihood to be compromised. A password length that is fewer than 16 characters is calculated using both alphanumeric characters and hexadecimals. Sixteen-character length passwords are generated using hexadecimals only.

While password strength is important, because a strong password is less likely to be compromised, so is usability. A long and complex password can be difficult to use, so administrators also have the ability to choose a shorter password length.

In addition, administrators can now configure a custom message to help walk users through ID recovery.

1.2.8 Single sign-on (SSO) and name mapping in Domino

The cookie, or LTPA token, that is created to authenticate users for single sign-on includes the name of the user who has been authenticated. When Domino creates the LTPA token, it places the Domino distinguished name in the LTPA token by default.

If an IBM WebSphere® server obtains the LTPA token from a user trying to access the server, the WebSphere server must be able to recognize this name format. If it does not, the LTPA token is ignored, single sign-on fails, and the user is prompted to log in again.

Additionally, users have been obliged to first login to WebSphere in multi-identity scenarios with Domino, because WebSphere has had trouble in multi-identity scenarios accepting LTPA tokens containing a Domino-format name. There has been a suboptimal workaround allowing Domino names: namely to write a WebSphere custom login module code. Accordingly, SSO works as long as WebSphere creates the LTPA token. Domino can accept LTPA tokens created by WebSphere.

Therefore, in Release 7.0, the Domino administrator can configure the user name to put into an LTPA token that Domino creates. This can be done manually to configure the name in a new directory field. It can also be done to programmatically configure the name in a directory field (for example, assisted by a directory synchronization tool such as IBM Tivoli® Directory Integrator).

The configuration should provide a name that will be accepted by WebSphere. The net result being that, now, users can log in to either Domino or WebSphere first with the same results.

2

Custom password policies in Lotus Notes and Domino 7

Passwords are at the forefront of the never-ending battle to keep systems secure. After all, the human element is generally the weakest link in the mesh of security measures and functionalities put in place to secure systems, because users are just not good at remembering passwords or can be duped, through social engineering, to divulge their password.

Having identified early on such problems with user passwords, Lotus implemented over time a number of password-related measures, such as the changing hieroglyphs in the logon window for Lotus Notes, the password quality scale, and password checking.

In Notes and Domino 7, Lotus has extended the policies and policy settings functionality to manage specifically passwords as part of a comprehensive security policy settings configuration.

In this chapter, we describe new functionality introduced in Notes and Domino 7 that helps control and administer passwords more effectively, enhancing the overall security of the Notes and Domino infrastructure while being easy on users.

We summarize what policies and policy settings documents are and how these have evolved in Notes and Domino 7. Then, we discuss custom password policies, explaining how to implement and manage them, including best practices and things that should not be done with them. Finally, we discuss other new features and functionality related to passwords in Notes and Domino 7.

2.1 Policies and policy settings

Because the main topic covered in this chapter deals with policies, let us take a moment to review what these are so as to better cover the new functionality introduced in Release 7.

Do not confuse Domino policies with corporate security policies. These are two completely different things.

A corporate security policy is a set of guidelines and standards used in an organization to establish and enforce secure information practices.

Domino policies, in contrast, permit administrators to control key aspects of the Notes and Domino infrastructure, specifically, how users work with Notes.

2.1.1 Policy basics

Specifically, a policy is a document that identifies a collection of individual policy settings documents. As shown in Figure 2-1, the policy settings documents cover six administrative areas: registration settings, setup settings, desktop settings, mail settings, mail archiving settings, and security settings.

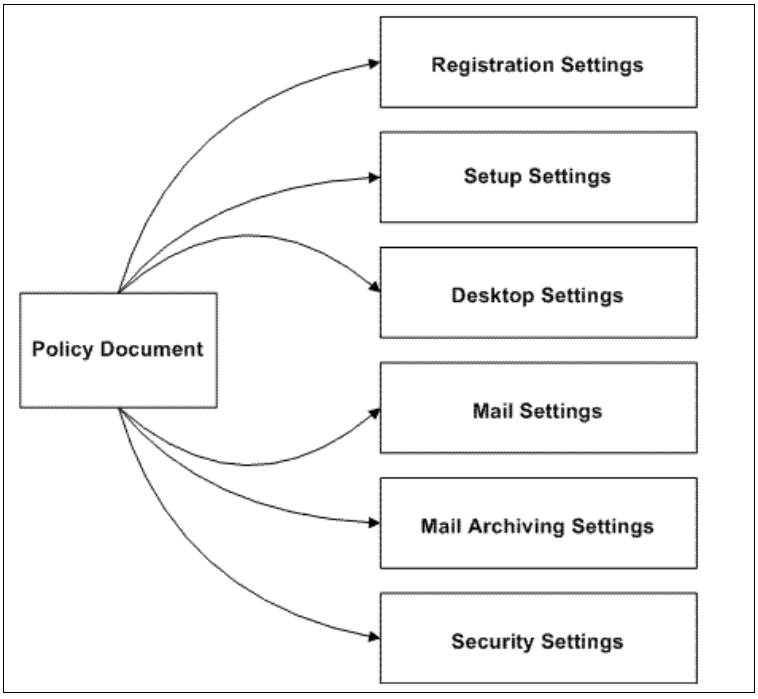


Figure 2-1 Policies and policy settings documents

These areas are best described as follows:

- Registration: If a policy including registration policy settings is in place before the administrator registers Notes users, these settings set default user registration values including the user password, Internet address format, roaming user designation, and mail.
- Setup: If a policy including setup policy settings is in place before the administrator sets up a new Notes client, these settings are used during the initial Notes client setup to populate the user's Location document. Setup settings include Internet browser and proxy settings, applet security settings, and desktop and user preferences.

- Desktop: Administrators can use desktop policy settings to update the user's desktop environment or to reinforce setup policy settings. For example, if a change is made to any of the policy settings, the next time users authenticate with their home server, the desktop policy settings restore the default settings or distribute new settings specified in the desktop policy settings document.
- Mail: Administrators can use mail policy settings to set and enforce client settings and preferences for mail and for Calendaring and Scheduling.
- Mail archiving: Administrators can use archive policy settings to control mail archiving. Archive settings control where archiving is performed and specify archive criteria.
- Security: Administrators can use security settings to set up administration execution control lists (ECLs) and define password-management options, including the synchronization of Internet and Notes passwords.

Each of these policy settings documents defines a set of defaults that apply to the users and groups to which the policy is assigned. After a policy is in place, it is possible for administrators to easily change a setting, and it will automatically apply to those users to whom the policy is assigned.

2.1.2 Organizational and explicit policies

There are two types of policies: organizational and explicit. It is important to understand the differences between the types; otherwise, it might lead to an improper implementation of these policies. In addition, there are exceptions that you can apply to these policies.

Organizational policies

An organizational policy automatically applies to all users registered in a particular organizational unit. For example, if an administrator wants to see default settings distributed to all users at the fictional ITSO Acme company registered in Sales/Acme, the administrator simply creates an organizational policy named */Sales/Acme. Then, when that same administrator uses the Sales/Acme certifier ID to register a user, that user automatically receives the settings in the corresponding organizational policy.

If a user is moved within the hierarchical structure (for example, because the user transfers from the sales department to the marketing department), the organizational policy for the corresponding certifier ID is automatically assigned to the user. For example, if the administrator moves the user from Sales/Acme to Marketing/Acme, all settings defined in the desktop, archiving, and security policy settings documents associated with the */Marketing/Acme organizational policy are assigned to the user. The new policy settings become effective the first time users authenticate with their home server.

Explicit policies

An explicit policy assigns default settings to individual users or groups. For example, to set a six-month certification period for contract workers in all departments, the administrator simply creates an explicit policy and then assigns it to each contract employee or to the group that includes all contract employees.

There are three ways to assign an explicit policy: during user registration, by editing the user's Person document, or by using the Assign Policy tool.

Using exceptions

It is possible for administrators to assign an exception attribute to either an organizational or explicit policy.

An administrator uses exceptions to allow the user to override a policy setting that is otherwise enforced throughout an organization. When an exception policy is created, the administrator can specify only the settings that will not be enforced. Then, when the administrator assigns the exception policy, it exempts users from enforcement of those settings only.

Exception policies are a way to give someone in an organization special treatment, possibly because of their position or job requirements. For example, the */Acme policy includes a Registration policy setting that enforces a mail database quota of 60 MB. However, a small group of employees in Acme need to exceed this quota. The solution is to create an “exception” policy that includes only a Registration policy settings document that does not set a quota limitation on the mail database. When this exception policy is assigned to users, they can override the database quota setting. Because exception policies defeat the enforcement of policy settings, use them sparingly.

Order of application

Figure 2-2 shows the order of application of organizational policies, explicit policies, and exceptions, where exceptions have the highest priority and organizational policies have the lowest priority.

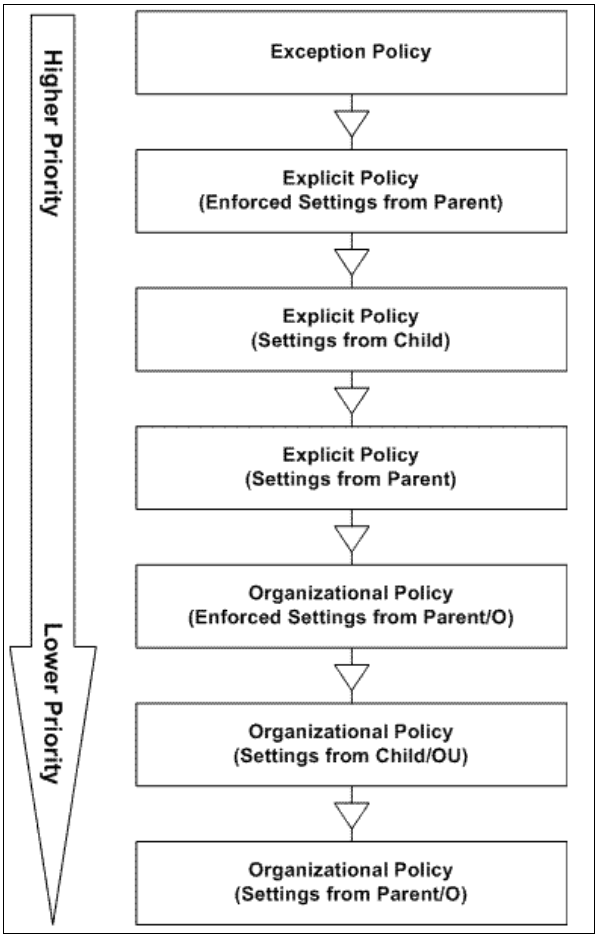


Figure 2-2 Order of application of organizational policies, explicit policies, and exceptions

2.1.3 Policy hierarchy

The effective policy for a user is a set of derived policy settings that are dynamically calculated at the time of execution.

The field values in an effective policy can originate from many different policy settings documents. Each hierarchical level can have an associated policy, so users can have a combination of policy settings that include the values set at their OU level and those inherited from a parent policy. The resolution of those settings, stepping up through the organizational hierarchy, determines the effective policy for each user.

In addition to organizational policies, users can also have explicit policies assigned to them. In that case, the order of resolution is that all organization policy settings are resolved first, and then any explicit policy settings are resolved.

For example, if an administrator wants all users to use the same Internet mail name format, the administrator would set that value in the Registration policy settings document for the top-level policy. After the administrator has set this value, it does not need to be changed or reentered in subsequent child policies. This value is simply “inherited” from the parent by having the inherit option selected. However, if there is a select group of international users for whom this setting is a problem, it is then possible for the administrator to create an explicit policy that applies to the select group only. The combination of the explicit and organizational policies together provides the control and the flexibility required.

Figure 2-3 on page 12 shows a flow chart that explains how this all works. In addition, there are two tools that help you determine the effective policy governing each user. The Policy Viewer shows the policy hierarchy and associated settings documents, and a Policy Synopsis report shows the policy from which each of the effective settings was derived.

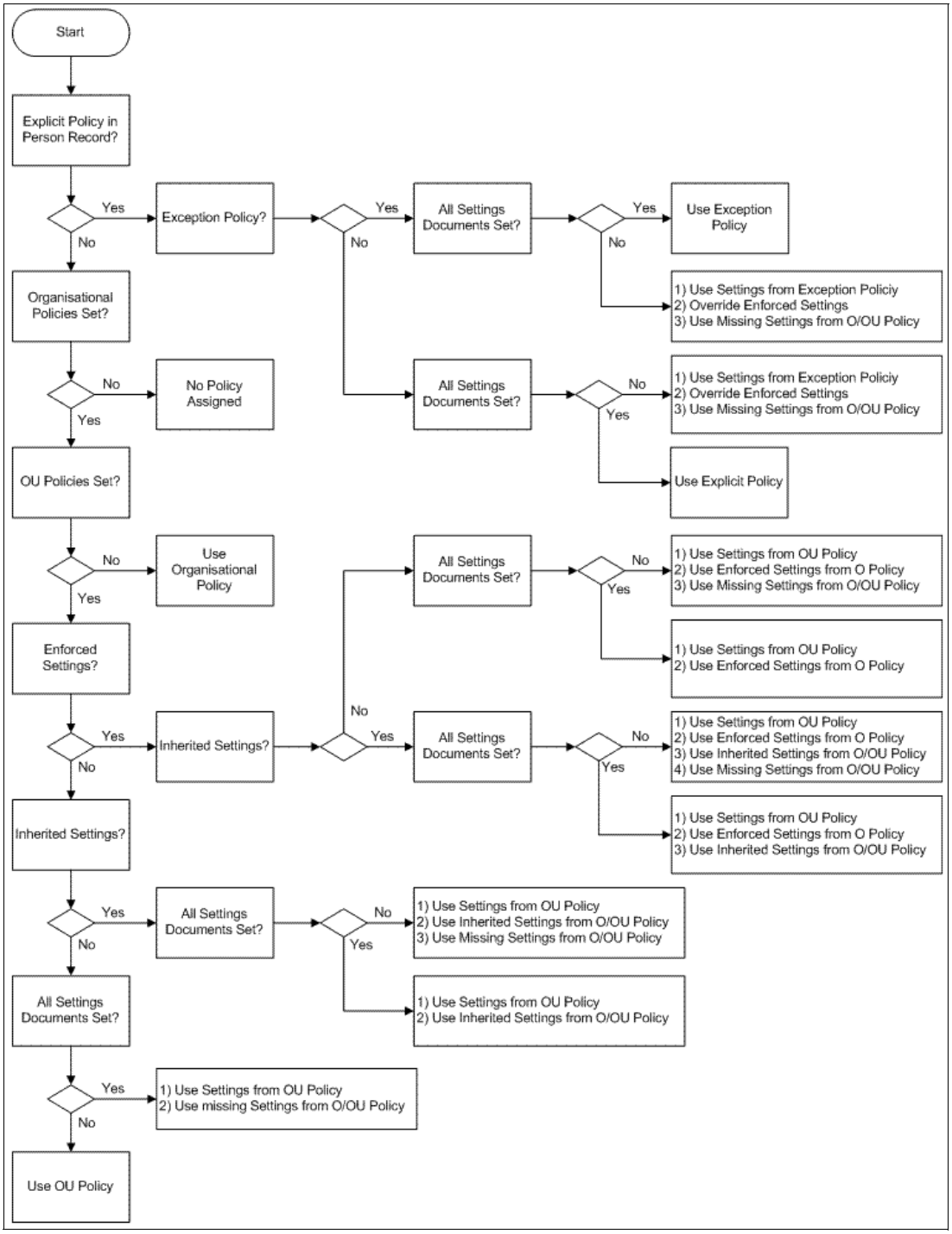


Figure 2-3 How policies work and are interpreted

Inheritance and the child policy relationship

Inheritance plays an important role in determining a user's policy settings in both organizational and explicit policies. Through the parent-child relationship, it is possible for the administrator to create a hierarchy of policies to set the desired administrative practices across the enterprise.

In a policy hierarchy, policy documents build the relationship, and policy settings documents determine the value of the fields based on their position in the hierarchy. Using field inheritance and enforcement, the administrator controls the default settings.

In organizational policies, the hierarchy of policies is determined automatically based on the organization's hierarchy. The policy */Sales/Acme is the child policy of */Acme. Because explicit policies do not follow the organizational structure, when the administrator creates explicit policies, the administrator builds in the hierarchy, based on the naming structure. For example, the administrator creates an explicit policy named /Contractors that includes several settings that apply only to contract employees who might be employed for six month to a year. However the administrator wants short-term temporary employees, employed for only one or two weeks, to inherit only some of those settings. The administrator then creates a child explicit policy called Short term/Contractors.

We are now ready to discuss custom password policies.

2.2 Overview of custom password policies

Now that we have covered the policies and policy settings basics, let us focus once again on passwords. The new features in Notes and Domino 7 address passwords for the needs of both administrators and end users.

Indeed, both groups need more and better password management, because there is a struggle that exists between end users and administrators, with each group being at opposite ends regarding passwords.

Passwords are sometimes difficult for end users. These passwords are what give authorized users access to systems and applications such as Lotus Notes, while at the same time keep unauthorized users from accessing their account, their mail, and other pieces of confidential information. Therefore, most users see that passwords are a good thing. However, given that users are notoriously bad at remembering passwords, they will be unhappy with passwords that are not easy to remember, upset that they cannot choose their passwords such as their birth date or the name of their favorite animal, and frustrated that they are prevented from writing these passwords down and putting them under their keyboard. From the point of view of users, passwords would only be a small, easy formality, no more complex than bob, sailing, password, or notes.

Passwords can be difficult for administrators as well. They need to ensure that passwords are sufficiently complex so that the systems these passwords protect are indeed well protected (Lotus Notes in particular, because this is the messaging system of the organization and serves as the backbone of the business that needs to be conducted on a daily basis), that the passwords are not easily guessable, and that unauthorized people are efficiently prevented from using unauthorized accounts. From the point view of administrators, passwords would a complex set of mental challenges testing the real mettle of users, confirming the profound desire of these users that they indeed want to access Lotus Notes, and demanding top quality passwords, such as “puttingthe*I*innotesID” and “stream&pond#river%lake!sea$ocean”.

Therefore, this leads to a loud and ever-continuing litany of complaints by users who feel they are being asked to remember passwords that are too complex and administrators that understand why the users are complaining but would like nothing better to make these passwords even more complex.

The aggravation users are experiencing with passwords is no doubt inflamed by the fact that many organizations are beginning to implement corporate security policies for password length and complexity. To add to the administrators’ challenge, recent changes to information

protection and data privacy laws, such as the Italian Privacy Act, include specific requirements for the selection of secure passwords that need to be enforced on an organizational basis.

There are no miracle cure for making users happy, making administrators relax their stance on passwords, and making the systems comply with new regulations in place (either corporate, dictated by the security people, or legal, dictated by new laws). However, the new features in Domino 7 promise to make at least Notes password management easier.

Namely, the custom password policy feature was added to help administrators configure and enforce specific password parameters for the selection of secure passwords. At a minimum, these policies ensure that passwords are not easily predictable, which represents a good security practice no matter the case. At a maximum, they can be used to help organizations enforce password requirements that will fit almost any set of corporate or government security requirements and perhaps find passwords that help create a happy middle ground for all the involved parties.

Note: Custom password policies apply only to Notes passwords. They do not apply to Internet passwords, such as those used for Domino Web Access.

Custom password policies are created and applied through a security policy settings document. Through a custom password policy, administrators can specify, restrict, or prohibit any of the following items in user passwords:

- User name as part of the password

  If the user's name is John Doe, for example, you can use a custom password policy that restricts or prohibits users from using passwords such as johndoe456, jdoe123, or johnd789.

- Repeating characters

  If the user decides to use a repeating character trick, you can put in place a custom password policy that restricts or prohibits users from using passwords such as 122333444455555, wooooow, or even jb00007.

- Unique characters

  There can be reasons for an organization to discourage the use of special characters so that passwords conform uniformly to systems throughout the organization. For example, all systems can handle basic 7-bit ASCII characters, while some might not understand anything beyond that (which is not the case for Lotus Notes, because it can handle all Unicode characters), or as another example, some characters represent control characters and thus should be restricted in their use to avoid problems. Therefore, you can use a custom password policy that restricts or prohibits users from using passwords such as sérénité, john§doe, or jb007!

- Use of special characters (such as punctuation), numbers, and uppercase and lowercase characters

  In an effort to prevent brute-force dictionary attacks, many organizations have a password policy that requires the use of at least one special character, one number, or the use of mixed-case characters. You can put in place a custom password policy that enforces the organization's password policy and requires users to use passwords such as Me&Myself&1 or Me?You?The2ofUs. Specifically, the custom password policy can specify that:

  - Passwords start or end with certain character types.
  - Passwords use combinations of mixed-case characters and non-alphabetic characters.

– Password length and strength maximums or minimums.

In a custom password policy, administrators can also require that users change their password on first use.

The other tremendous advantage of custom password policies is the ability to establish different password requirements for different types of users, based on the group to which they belong, where they are in the certifier tree, or, where warranted, based on exceptions based on organizational or legal security requirements. For example, the finance group might need a very stringent custom password policy in order to be compliant with government accounting regulations, while the marketing group might not require one at all.

Finally, note that some of the abilities afforded by Lotus Notes and Domino to help users manage passwords themselves (for example, expiration, synchronization, grace period, and password history) have existed in security policies since Domino 6. The primary purpose of the Domino 7 custom password policy is to specifically define the makeup of the password itself, that is, its strength, length, and complexity. This said, however, understand that there are limitations regarding custom password policies, in that custom password policy settings do not:

- Support random password generation, either through user registration or the User Security window. If a custom password policy is in place, users must enter a password when prompted to do so.
- Apply to IDs protected with multiple passwords.
- Apply to IDs protected with smartcards.

2.3 Implementing password policies

Custom password policies are downloaded to the Notes ID file when users first authenticate through their Lotus Notes to their Domino home server.

After it is stored in the ID file, the policy settings apply to the user's password the next time that user logs in to the Notes client. If the policy specifies that the password must be changed on first use, the user will be prompted to do so at that time.

Custom password policies can be applied at user registration, or to users or groups of users that have already been registered. Note that the requirement to change password on first login can only apply to new users who have had the policy applied to them at registration. If the policy has been applied to a new user, the user must first authenticate with the server in order to be prompted to change password first use. If a custom password policy is applied to users who are already registered, these users will not be required to change their passwords when they log in after the policy has been applied.

If a password policy is implemented, Domino enforces its use. If the user does not change the password to conform to the policy, or cancels out of the change password window, the user receives an error message stating that the password does not meet policy requirements, and the Notes client shuts down.

Important: Domino does not have many validation checks for custom password policies.

It is possible for an administrator to create a policy such that no password will ever meet the requirements (for example, maximum length = 4, minimum password quality = 8). If such a password is inadvertently implemented, it will not work. It is impossible for a password to be implemented, the user will not be able to change the password, and thus the Notes client shuts down. Administrators need to make sure that the password policies they implement make sense and can actually be implemented.

SPR JCAL68TP9Z addresses this. It is an enhancement request for a “sanity button” that will verify that numbers entered in custom password policy fields match specified character minimum and maximum totals.

The workaround is twofold. First, the administrator needs to be clear about the requirements for a password policy and to make sure that one requirement does not conflict with another. Second, as with all things being newly implemented, it is important to test the customer security policy in a quality assurance or test environment to see if the actual behavior of the custom password policy matches the expected one.

2.3.1 Custom password policy configuration

The term “custom password policy” is a bit misleading. It is not, in itself, a separate policy document. It is part of the Security Settings form. However, there might be a bit of confusion at first when trying to create and configure a custom password policy, because the Password Management tab on the Security Settings form looks like the window shown in Figure 2-4 on page 17.

The odd thing about the form is that there is a sub-subtab, the Password Management Basics tab, which hints at the fact that there should be additional tabs there (after all, the designers at Lotus are pretty logical about their design choices and do not put things where it would not make sense to put them). That is correct: The Custom Password Policy tab is unavailable.

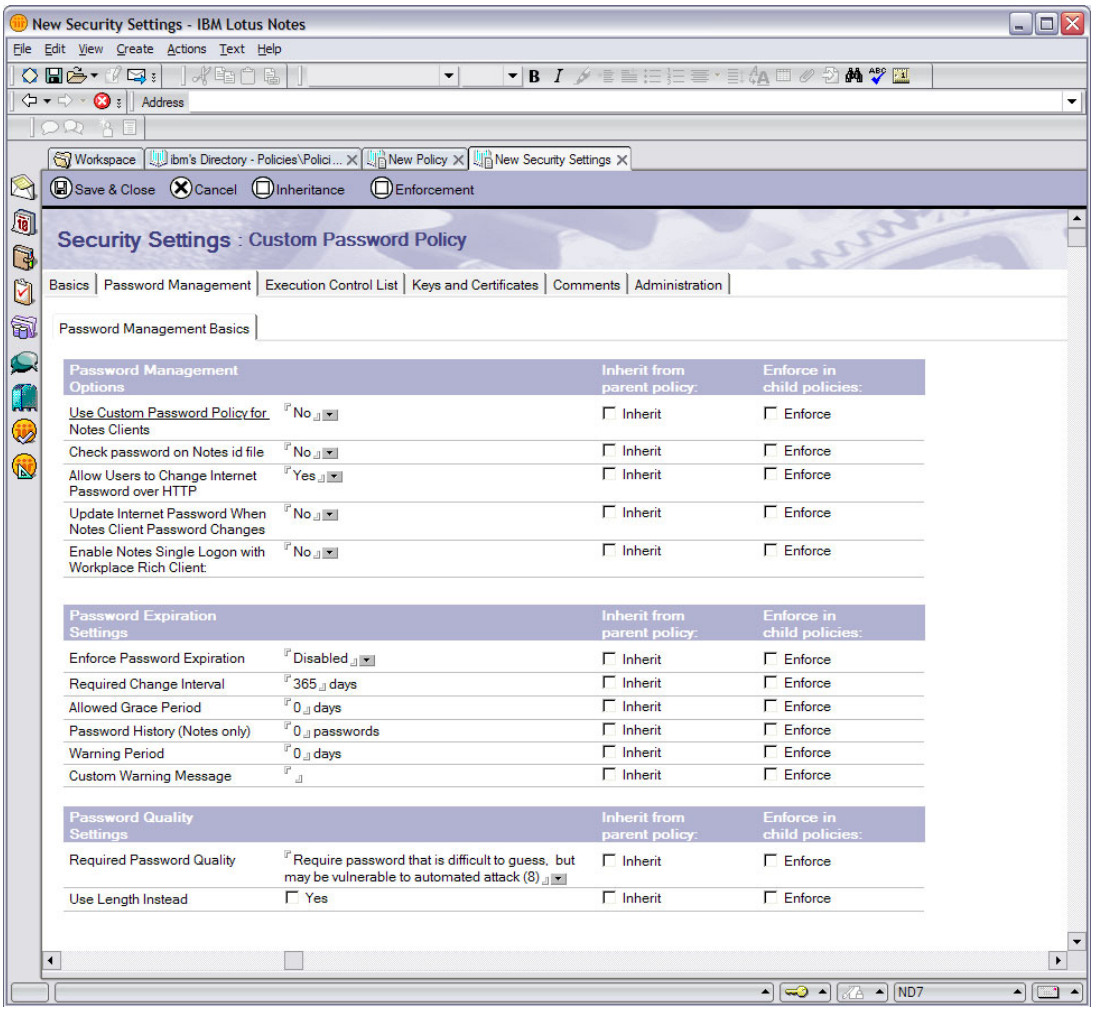


Figure 2-4 The Security Settings form, Password Management tab

To display the Custom Password Policy tab, change the value of the “Use Custom Password Policy for Notes Clients” from No to **Yes**. This is rather dynamic in nature, because the tab appears when the value of the field changes to “Yes,” as shown in Figure 2-5, and disappears when the value of the field changes to “No.”

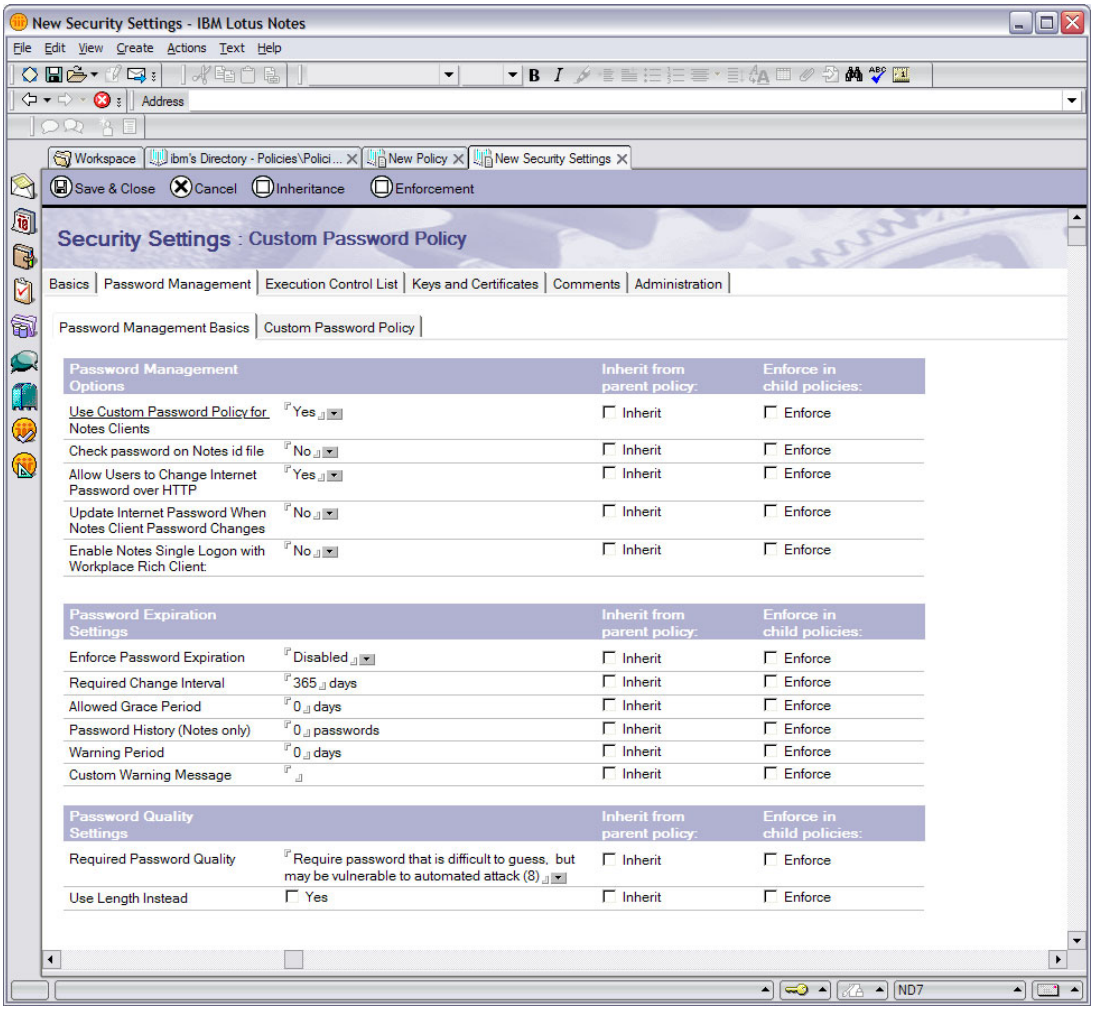


Figure 2-5 The Security Settings form, Custom Password Policy sub-subtab

The Custom Password Policy tab shows the settings for custom password policies, as shown in Figure 2-6 on page 19.

Note: Any custom password policy settings enabled for password quality and length override the password quality settings on the Password Management tab.

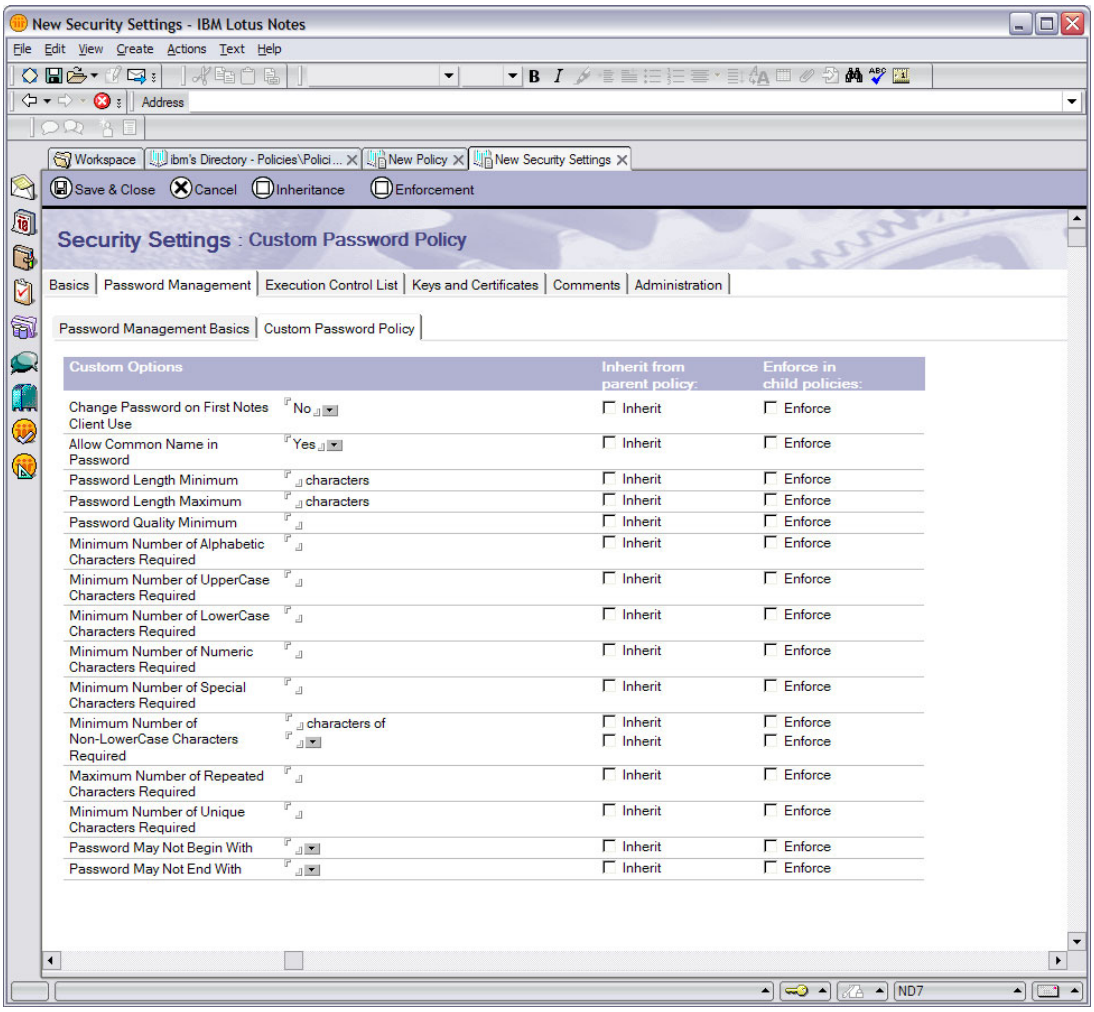


Figure 2-6 The Security Settings form, the settings on the Custom Password Policy tab

Custom password policy settings

Now that we know where in the Domino Directory to find these settings, let us take a closer look at them.

To configure a custom password policy, complete any or all of these settings. Again, it is important to remember that the requirements have to make sense. For example, specifying a password length maximum of 4 is not compatible with a password quality minimum of 15.

Here, we provide a detailed description of every setting on the Custom Password Policy tab of the Security Settings form:

- Change Password on First Notes Client Use

  This setting enables the administrator to force users to change their passwords the first time they log in using Lotus Notes. The setting applies only to newly registered users that have never logged in to Lotus Notes before. For existing users that have already logged in once, this setting does not apply.

- Allow Common Name in Password

  This setting enables the administrator to allow a combination of the common name of user to be used in passwords. For example, Frederic2517 is the password for user CN=Frederic Dahm/OU=Lotus/O=IBM, where the common name is Frederic Dahm.

- Password Length Minimum

  This setting enables the administrator to specify the minimum number of characters that users can have in their passwords. For example, if the administrator enters 5 in this field, the password will have to have at least 5 characters, regardless of its complexity (in other words, the password sandy fulfills the requirement but the password 7$up does not).

- Password Length Maximum

  This setting enables the administrator to specify the maximum number of characters that users can have in their passwords. For example, if the administrator enters 12 in this field, the password will not be allowed to have more than 12 characters, regardless of its complexity (in other words, twelverchars will be accepted but thirteenchars will not be accepted).

- Password Quality Minimum

  This setting enables the administrator to specify the minimum password quality value that users can have for their passwords. The concept of password quality value is a bit difficult to understand in terms of what a given level will require in terms of a password that will be compliant. Table 2-1 offers an example for each password quality level. It is worth pointing out that a password of quality level 7 will fulfill the requirements if the minimum password quality is set at 5.

Table 2-1 Password Quality Minimum

Password quality	Password
3	dog password pwd 2d4
4	5786 atof r2d2
5	d0gs doGs scAle
6	sCa1e dogcat pw46wp
7	cat7dog catSrOK
8	tyughvbn one21two rt 7uj
9	one2 1two onetwothree
10	pAssw0rd pwd46dwp
11	winD39_BP the way we were
12	PwD46dWp rtyughjkbnml GoneWithTheWind

Password quality	Password
13	Gone With The Wind 4891spyONu
14	tree forest grass rock thedogisontheporch
15	thecathidesunderthebed tdiotp&tchutb
16	thecowjumpedoverthemoon thedishranawaywiththespoon stream8pond1river7lake2oceanz

- Minimum Number of Alphabetic Characters Required

  This setting enables the administrator to specify the minimum number of alphabetic characters that users are allowed to have in their passwords. If the administrator enters 5 in this field, for example, the password fr3d3r1c will be considered valid, but the password s4ndy will not.

- Minimum Number of UpperCase Characters Required

  This setting enables the administrator to specify the minimum number of uppercase characters that users are allowed to have in their passwords. If the administrator enters 3, for example, the password WAMozart will be considered valid, but the password LotusNotes will not.

- Minimum Number of LowerCase Characters Required

  This setting enables the administrator to specify the minimum number of lowercase characters that users are allowed to have in their passwords. This is the same as the preceding setting but with lowercase characters instead of uppercase characters.

- Minimum Number of Numeric Characters Required

  This setting enables the administrator to specify the minimum number of numeric characters that users need to have in their passwords. If the administrator enters 3, the password LotusNotes7.0 will not be considered valid, but the password LotusNotes654 will be considered valid.

- Minimum Number of Special Characters Required

  This setting enables the administrator to specify the minimum number of special characters, namely punctuation, that users need to have in their passwords. For example, If the administrator enters 2, the password Lotus!Notes will not be considered valid, but the password custom%password*policies will be considered valid.

- Minimum Number of Non-LowerCase Characters Required

  This setting enables the administrator to specify the minimum number of special characters, numbers, and uppercase characters that is required in user passwords. A higher value here makes passwords more difficult to guess.

  After a number is entered, a checklist opens, listing the character types that can be specified for this requirement. It is possible to pick any combination of the following values:

  - Numbers
  - Special characters
  - Uppercase

  If the administrator enters 2 and all three types are selected, Lotus!Notes7%0 will be considered valid.

- Maximum Number of Repeated Characters Required

  This setting enables the administrator to specify the maximum number of repeated characters, of any kind, that are allowed in user passwords. If the administrator enters 2, the password Sweet will be considered valid, but the password Wheee! will not be considered valid.

- Minimum Number of Unique Characters Required

  This setting enables the administrator to specify the minimum number of characters that appear only once in a password.

- Password May Not Begin With

  This setting enables the administrator to specify the type of characters with which passwords cannot begin. If the administrator enters lot, the password lotusnotes will not be considered valid, but the password ibmlotusnotes will be considered valid.

- Password May Not End With

  This setting enables the administrator to specify the type of characters with which passwords cannot end. If the administrator enters lot, the password pilot will not be considered valid, but the password lotusnotes will be considered valid.

Because all this might seem a bit complex, we illustrate these custom password policies using a couple of examples.

2.3.2 First example of a custom password policy

These examples involve two fictional corporations: ITSO Acme Corporation and ITSO Widget Corporation. The two companies do not agree about what kind of passwords users need to type in as part of the authentication process.

Acme has a liberal approach to passwords and has implemented a custom password policy with the following requirements, as shown in Figure 2-7 on page 23:

- Passwords must have a minimum length of 6 characters and a maximum length of 8 characters.
- Passwords must contain at a minimum of one alphabetical character and one non-alphabetical character.
- Passwords can have numeric characters in the first and in the last place.
- Passwords can have a maximum of two identical characters in a row.
- User IDs cannot be part of the password.

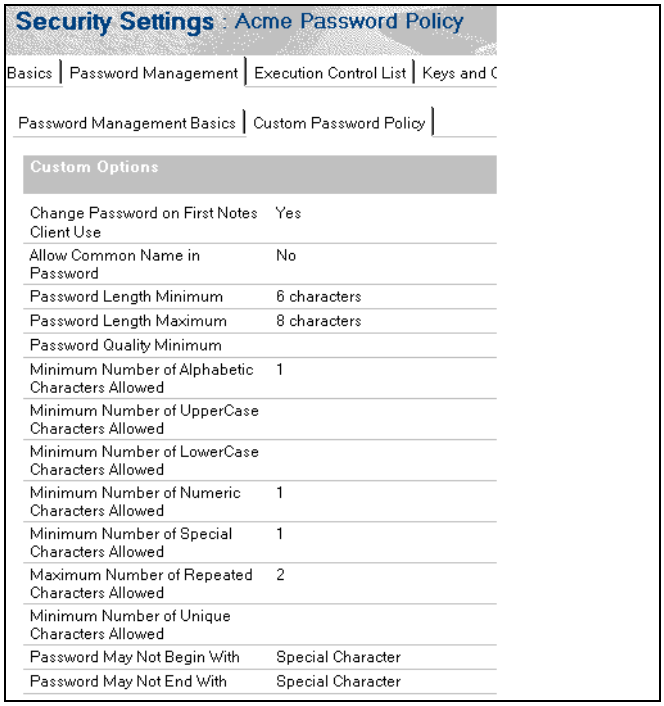


Figure 2-7 Example of a Custom Password Policy

The password management requirements of the policy include the following settings, as shown in Figure 2-8:

- Password change interval is 35 days.
- Password warning interval is 5 days.
- Password history is 12 passwords.

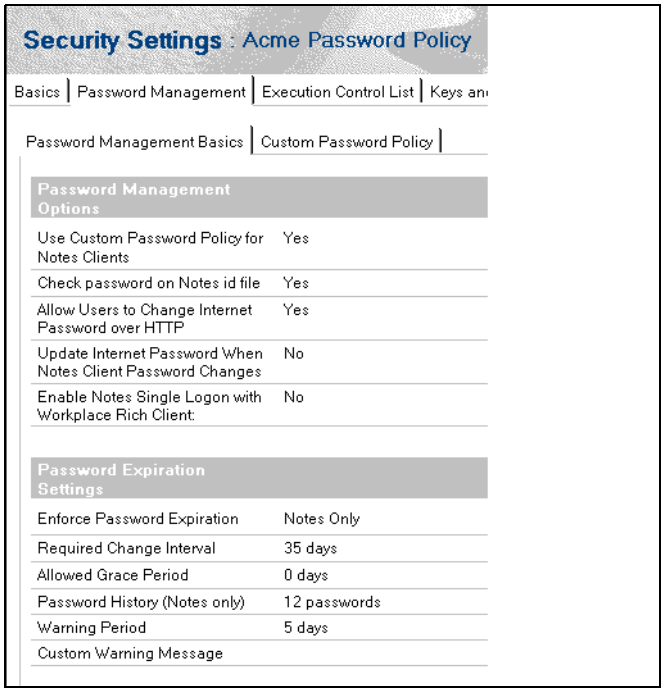


Figure 2-8 Details of a Custom Password Policy

2.3.3 Second example of a custom password policy

ITSO Widget Corporation has a more restrictive approach to passwords and has implemented a custom password policy with the following requirements, as shown in Figure 2-9:

- Passwords must have a minimum length of 15 characters and a maximum length of 15 characters (which means that users will have to use passwords or passphrases that are exactly 15 characters long).
- Passwords can have a maximum of 3 identical characters in a row.
- For numeric characters, passwords can have numeric characters in the first and last place. In addition, there can be no less than 2 numeric characters in the password.
- Passwords can have special characters in the last place, but not in the first place. In addition, there can be no less than 2 special characters in the password.
- Case is also important, because passwords must have a mix of uppercase and lowercase characters. A minimum of 2 lowercase characters is required and a minimum of 2 uppercase characters is required.
- As with Acme Corporation, user IDs cannot be part of the password.

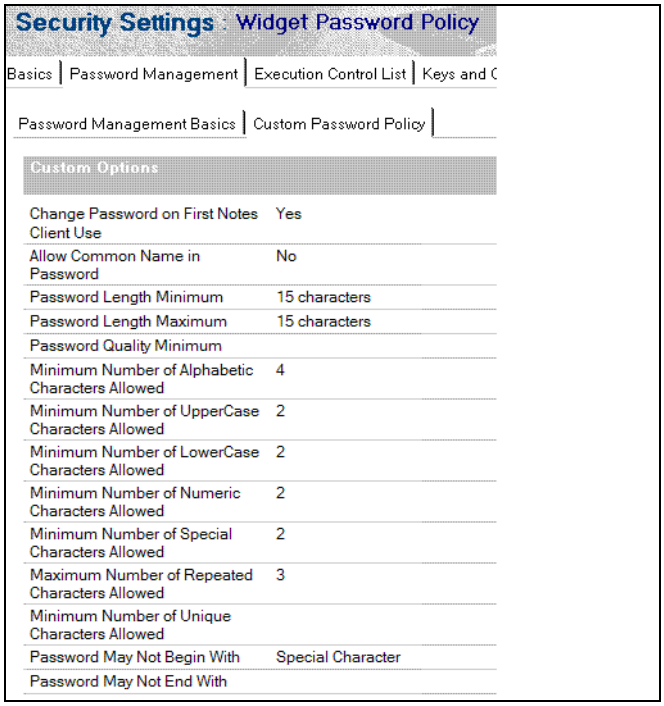


Figure 2-9 Example of a Custom Password Policy

The password management requirements of the policy include the following settings, as shown in Figure 2-10 on page 25:

- Password change interval is 21 days.
- Password warning interval is 5 days.
- Password history is 15 passwords.

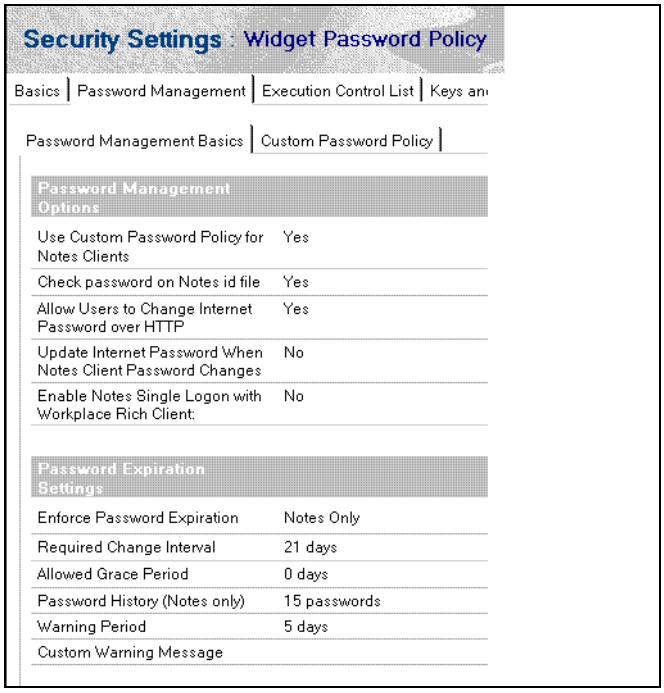


Figure 2-10 Details of a Custom Password Policy

2.3.4 Additional new password management policy settings in Release 7

This chapter would not be complete without mentioning that password management settings are independent of custom password policies. It is possible to configure these settings in a security policy without implementing a custom password policy.

Password expiration warning period and message

There are two new security policy settings in Domino 7 that enable administrators to fine-tune password expiration: Warning Period and Custom Warning Message.

The Warning Period setting enables administrators to specify, in days, the time prior to password expiration at which the user receives an expiration warning message (that is, “Your password will expire in 10 days”). The default for this setting is 0, meaning that users will get no warning as to when their passwords expire. However, if password expiration is enabled and the required password change interval (the number of days a password is valid before it must be changed) is set to less than 30 days, the value of this field will be calculated automatically to be two-thirds of the password change interval. If the value is calculated, it cannot be overwritten.

Administrators can now also specify a Custom Warning Message to users whose password has passed the expiration threshold specified in the Warning Period field. Prior to Domino 7, users only received a standard error message.

The Custom Warning Message setting is for Notes clients only, regardless of whether password expiration is enabled for both Notes and Internet passwords or Internet passwords only. Internet users do not see the warning message.

Single login with IBM Workplace Collaboration Services

IBM Workplace Collaboration Services 2.6 Managed Client has the option of using a Notes plug-in, through which IBM Workplace users can access the Notes databases and

applications. Domino administrators can enable the ability, in a security settings document, for Workplace users to specify that the Workplace client “remembers” the Notes password when accessing Notes applications in order to avoid having to be repeatedly prompted for their Notes passwords when attempting to access Notes applications.

This option can only be configured through a security settings policy document. Moreover, it is just an option. Workplace users are not required to use it. Rather, they chose to enable the option the first time they attempt to access Notes when they are presented with the Notes User Security window.

This completes our review of the new security features provided by custom password policies. The next chapter covers ID recovery.

3

ID recovery enhancements

The Notes ID is the cornerstone of the Lotus Notes and Domino security model, because it is used at the core of the Notes public key infrastructure (PKI). The Notes ID is a small file (meaning that it is only a few kilobytes in size) that contains many things that are necessary to use the services provided by the PKI built into the Notes client (and Domino server).

In this chapter, we revisit the Notes ID file, explaining its variants and explaining what can be done to recover from a situation in which the file can no longer be used, whether because the file is corrupted (a rare occurrence), the file was lost (for a number of reasons, another rare occurrence), or because users have simply forgotten the password that encrypts their Notes ID file (which covers most of the cases requiring the information contained in this chapter).

No matter what the reason for no longer being able to use a Notes ID, this has serious consequences. *Without their Notes IDs, users cannot access servers, nor read messages, nor access any data that they encrypted with the lost ID.* The Notes ID is that essential to accessing Domino servers and encrypted data.

While administrators are advised to urge users to keep backup copies of their Notes ID files in a secure place, which is referred to as an escrow mechanism and can be, for example, the storage of Notes ID files on a disk stored in a locked area, this addresses only the problem of lost or damaged ID files. A far more common problem is that of the *forgotten* password. The best tactic for preventing problems that occur when users lose or damage ID files or forget passwords is to set up Domino to recover ID files.

By the end of this chapter, you should understand the mechanisms for ID recovery and how these were improved and augmented in Release 7.0 of Notes and Domino.

3.1 Overview of the Notes PKI and Notes IDs

Before we can discuss how to recover Notes ID files, it is worth a little bit of time, and a few paragraphs, to review the Notes PKI and Notes IDs. This can help you understand the reason why Notes IDs play a significant role in the Notes PKI and why losing a Notes ID can completely prevent a user from using Notes and Domino.

It is not the goal of the present chapter to provide a detailed and complete explanation of public key infrastructures (PKIs), standard PKI implementations, and how the Notes PKI is implemented. If you are not familiar with these concepts, we encourage you to consult the previous publication in this series, *Lotus Security Handbook*, SG24-7017. However, we review some key elements about the Notes PKI to ensure a good comprehension of the Notes ID file and also the nature of the recovery mechanisms covered in this chapter, so as to avoid needless back and forth between the present publication and previous one.

3.1.1 Registration and certification

Before we describe the PKI natively present in Notes and Domino, it is important to talk about registration and certification, because these are frequently confused terms.

Registration

Registration is the action by which a user's details are entered in a directory. The directory in question is the Domino Directory. The work product of registration in Notes and Domino is the Notes ID.

Certification

Certification has two meanings that are pertinent to this chapter and to Notes and Domino. To certify is to confirm formally that something is true, accurate, and genuine and that it meets a standard. To certify is also to issue a license or certificate. The work product of certification in Notes and Domino is the creation of Notes certificates and their inscription in the Notes ID.

3.1.2 Certification hierarchies

When Lotus was first introduced, it offered only one type of certification: flat certification. Release 3 of Notes introduced hierarchical certification. Both flat and hierarchical certification were supported, in that it was possible to generate flat and hierarchical certificates. With Release 5, it was no longer possible to perform flat certification; however, previously generated flat certificates are supported in Releases 5, 6.0, 6.5, and 7.0 for backward compatibility.

Because flat certification is definitively a thing of the past, we do not cover it here, preferring to spend the time on hierarchical certification.

Hierarchical certification

For hierarchical certification, the server and user IDs have only one organization certifier and optionally up to four layers of organizational unit certifiers under the organizational certifier.

When users or servers are registered with a hierarchical certifier, they receive a certificate signed by that hierarchical certifier and inherit the certification hierarchy of the layers above. To illustrate this, consider Figure 3-1 on page 29, which shows an organization named Acme subdivided into two organizational units, Canada and USA, each subdivided into three organizational units, under which, finally, users are registered and certified.

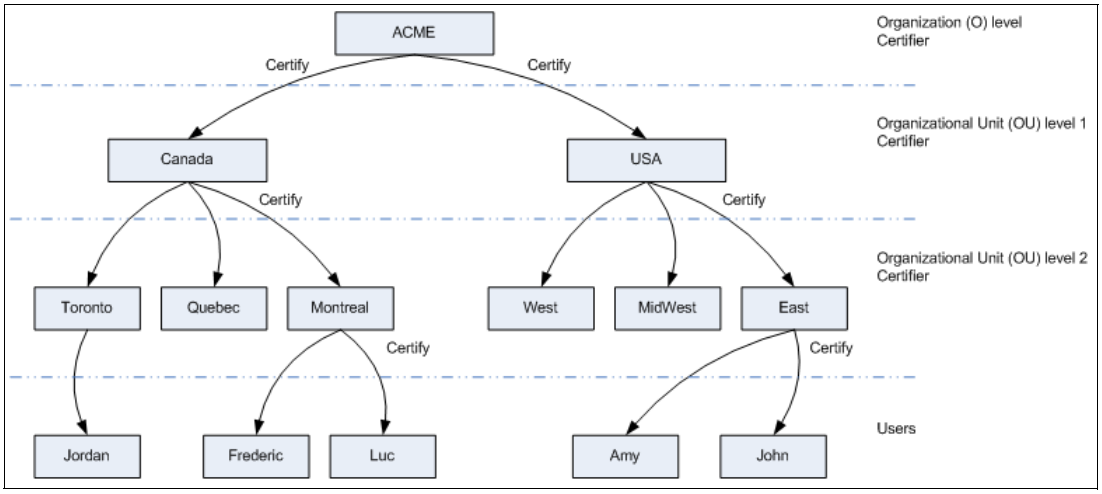


Figure 3-1 Hierarchical certification

When registering Amy as a new user, the administrator responsible for the East/USA/Acme organizational unit certifier registers her. One of the results of this process is a new, randomly-generated RSA private/public key pair. The administrator then creates a certificate for Amy by signing her new public key using the East/USA/Acme organizational unit certifier private key. As a result, Amy's user ID inherits the certification hierarchy of the East/USA/Acme organizational unit certifier.

In the case of Luc, it is very similar. When registering Luc as a new user, the administrator responsible for the Montreal/Canada/Acme organizational unit certifier registers him. One of the results of this process is a new, randomly-generated RSA private/public key pair. The administrator then creates a certificate for Luc by signing his new public key using the Montreal/Canada/Acme organizational unit certifier private key. As a result, Luc's user ID inherits the certification hierarchy of the Montreal/Canada/Acme organizational unit certifier.

Users and servers in the organization have fully distinguished names based on their certifiers. Each layer in the certification hierarchy inherits the fully distinguished name of the certifier used to create it and is, in turn, an ancestor to the layers below it.

In this example, the organization level certifier Acme has the fully distinguished name “o=Acme”. The organizational unit certifier Canada has the fully distinguished name “ou=Canada/o=ACME”. The organizational unit certifier USA has the fully distinguished name “ou=USA/o=ACME” and the organizational unit certifier Montreal has the fully distinguished name “ou=Montreal/ou=Canada/o=ACME”.

For Amy, her fully distinguished name is “cn=Amy/ou=East/ou=USA/o=Acme”. For Luc, his fully distinguished name is “cn=Luc/ou=Montreal/ou=Canada/o=Acme”.

When registering a server, the same applies, with the only difference being that a server ID is created instead of a user ID.

Finally, with regard to authentication, users and servers can authenticate with each other if they have at least one common ancestral certificate. In our example, this means that all users in the organization can authenticate with each other because they have the Acme certifier in common. Entities that do not share at least one common ancestor can still authenticate by going through a cross-certification process, which, for the sake of brevity, we do not cove here.

3.1.3 Notes ID

At the core of the Notes PKI is the Notes ID. The Notes ID is a small file (meaning it is only a few kilobytes in size), that contains many things that are necessary to use the services provided by the PKI built into the Notes client. We review these and cover the different types of Notes IDs in this section, giving you a base of understand for the explanation of Notes ID recovery later in this chapter.

Certifier, server, and user ID files

The Notes ID is essentially a “container” for certificates and encryption keys. There are three different types of Notes IDs:

- Certifier IDs

  These are Notes IDs that are used to generate other IDs. They come in two types: organization (O) certifier IDs and organizational unit (OU) certifier IDs. When IDs are generated, the organization certifier ID is created first; this is the master ID for the domain. This ID (if the organization is large enough) is used, in turn, to generate organizational unit certifier IDs. These certifiers are then used to generate the two other types of IDs: server IDs and Notes user IDs.

- Server IDs

  These are Notes IDs, as their name implies, that are used for servers that are part of the Domino domain. They uniquely identify every server in the domain.

- User IDs

  These are Notes IDs that are created for users who are part of the Domino domain. They uniquely identify ever user in the domain.

Because of their ability to generate user and server IDs, certifier IDs should be given more protection than the other types. Save these IDs on floppy disks and put them in a safe place, other than on the hard drive of the server. With Domino 6, it is possible to use the Domino 6 certificate authority (CA), which lets the administrator avoid circulating Notes certifier IDs for administrators to use.

Domino uses IDs to identify users and to control access to servers. The certifier, server, and user IDs contain the following information:

- The owner's name

  A user ID file can also contain one alternate name. A certifier ID can contain multiple alternate names.

- A permanent license number

  This number indicates that the owner is legal and specifies whether the owner has a North American, Internationa.l or Global license to run Domino or Notes.

- A pair of Notes certificates from a certifier ID

  Notes certificates are digital signatures added to a user ID or server ID. This signature, which is generated from the private key of a certifier ID, verifies that the name of the owner of the ID is correctly associated with a specific public key.

- Ancestral certificates

  There is a certificate for each ancestor certifier (at a minimum, one for the organization certifier and one for each additional organizational unit certifier).

- A private key

  Notes uses the private key to sign messages sent by the owner of the private key, to decrypt messages sent to its owner, and, if the ID belongs to a certifier, to sign certificates.

- (Optional) One or more secret encryption keys

  These are created and distributed by application developers or users with special privileges to a database to allow other users to encrypt and decrypt fields in a document.

- (Optional, Notes client only) Internet certificates

  An Internet certificate is used to secure SSL connections and encrypt and sign S/MIME mail messages. An Internet certificate is issued by a certificate authority (CA) and verifies the identity of the user. The user's private key associated with an Internet certificate is stored with that certificate.

Finally, the private key and the encryption keys in the ID file are encrypted using a key computed from the user's password so that only the owner can access it. Public information such as the user's name and public key are not encrypted.

Figure 3-2 illustrates the structure of a Notes ID, showing both the standard part that is created for every Notes ID and the optional part (which can be added to the Notes ID later).

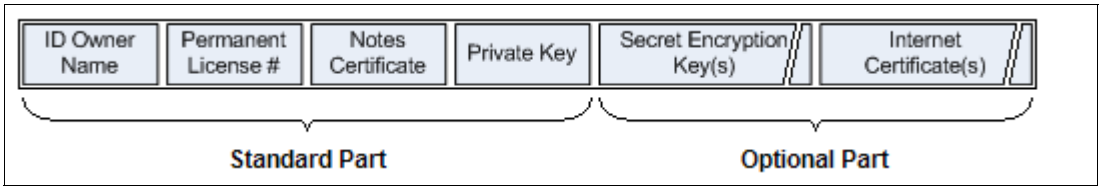


Figure 3-2 Structure of a Notes ID

Two things must be noted here:

- If a user is in the process of requesting a new private key or a name change, the pending information is also stored in the ID file. If a Notes private key is changed, the obsolete information is also stored in the ID file for backward compatibility (for example, you need the obsolete information to read old encrypted e-mail).
- There is some confusion on the part of certain users who download the Notes client, install it, and launch it. At that time, the client configuration process is started and a new Notes ID is generated for the user, which apparently does not require a certifier ID. This is a flat Notes ID that contains little information and will not be of any use the moment the Notes client tries to connect to a server in the domain.

Notes certificates

Lotus Notes authentication relies in large part on Notes certificates, which are stored in Notes IDs.

Casually speaking, a certificate is an electronic “stamp” that indicates a trust relationship among the entities in the Notes world. More formally, a certificate is a unique, digitally signed message added by a certifier to a Notes ID file that identifies a user or server. While the client can store and work with both Notes and Internet certificates, the rest of this section refers specifically to Notes certificates.

When a Lotus Notes user attempts to connect to a Lotus Domino server, whether it is a mail server or another type of Domino server in the organization, that person needs a certificate to identify himself or herself to that server, and the server needs a certificate to identify that person. Therefore, the Notes client and the Domino server involved in the authentication process present their certificates to each other. By examining the certificates, the Notes client will identify and authenticate the Domino server, and the Domino server will identify and authenticate the user.

In order to permit this trust relationship to be established, a number of pieces of information must be present in the certificates. A Notes certificate, like a Notes ID, contains a number of elements, such as:

- The name of the certifier that issued the certificate.
- The name of the user or server to whom the certificate was issued.
- A public key that is stored in both the Domino Directory and the ID file. Notes uses the public key to encrypt messages that are sent to the owner of the public key and to validate the ID owner's signature.
- A digital signature.
- The expiration date of the certificate.

The whole thing is then certified, meaning that it is digitally signed by the certifier using the certifier's private key in order to prove its authenticity. Figure 3-3 illustrates the structure of a Notes certificate within a Notes ID.

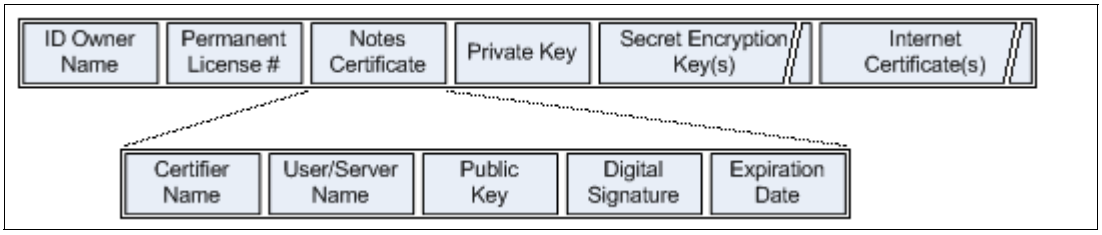


Figure 3-3 Structure of a Notes certificate

As mentioned, certificates are stored in Notes ID files. They are also stored in Person, Server, and Certifier documents in the Domino Directory. Given the nature of the contents of Notes ID files, it is best to think of them as being a kind of specialized database that stores Notes certificates and private/public key pairs. This database is then encrypted with the user's password.

When servers and users are registered, Domino automatically creates a Notes certificate for each server and user ID file. These Notes certificates have expiration dates, which means that a Notes ID must be recertified when its expiration date approaches.

In addition, if a user or server name changes, the corresponding Notes ID must be recertified so that a new certificate can bind correctly the public key to the new name.

Notes passwords

The main reason for having and using a Notes ID is for authentication. It is not the intention here to describe the complete authentication process with a Notes ID; this is best done by consulting the previous publication in this series, the *Lotus Security Handbook*, SG24-7017.

Let us just cover the basics of what a password provides and why a lost password can render a Notes ID completely unusable.

The password assigned to a Notes user ID during registration is a mechanism to protect the Notes ID file from unauthorized use. A Notes user attempting to use the Notes ID file will be required to enter the password for that Notes ID file.

There is some confusion in regard to the Notes ID password. It is used solely to unlock the Notes user ID file itself, and nothing more. It is the key pair contained in the ID that is actually used to identify the user.

Users can have more than one copy of their Notes ID file and these different copies can have different passwords. This basically means that to change the password, the user must know the existing password for each copy of the ID.

Although a Notes ID recovery feature was introduced in a previous version of Notes, it is still considered good practice to back up the ID files and to remember their passwords. However, in cases where this is not possible or where the security policy of the organization prevents such an escrow service to be set up, there will be instances where users forget their passwords and need help. This is where ID recovery comes in.

3.2 Notes ID recovery

In summary, as we said in the introduction to this chapter, you can use Notes ID recovery for either of the following situations: to replace a lost or corrupted Notes ID File, or to recover a forgotten password that encrypts the content of a Notes ID file.

For the rest of this document, we simply use “ID” for the term “Notes ID” because we explained this earlier and the association that an ID is a Notes ID should have been made by now.

3.2.1 Basic observations

Before delving into the fine details of ID recovery, there are some overall points that need to be made and understood:

- Where instructions are provided, these concern standard Notes certifiers and Notes user ID files, not Internet certificates or key rings.
- As with any new feature and new functionality introduced into a production environment, test the information provided first in a test or quality assurance environment. After it has been validated, conduct a small pilot with the help of willing Notes users. If the pilot is successful, a deployment of the new functionality can be planned and delivered to the production environment.
- Many of the Domino features outlined in this chapter (and in some of the other chapters as well) have inherent and variable time delays. For example, after creating a new user, the user sometimes cannot immediately log on, but must wait for the new ID to be registered by the CA process. When the list of recovery authorities in a certifier is changed, the new list is downloaded to the users’ ID files after a period of time. These delays can be several hours each. If it appears that a certain step of these instructions has failed, you should wait a half-day and try again.
- Related to the previous point, it is also worth noting that the commands **`tell adminp process all`** and **`tell ca show queue *`** do not always display or clear pending work items related to these features. The Domino CA process does some things silently on its own schedule.

3.2.2 Overview of ID recovery

The ID recovery process is based on Domino keeping an encrypted backup copy of the ID file in a designated mail in database to permit the recovery of an ID file that was lost of corrupted.

The backup copies of the ID files are encrypted using a random key. They cannot be used with Notes until they are recovered. They are stored as attachments in documents contained in a special Notes database set up specifically for ID recovery. This database is a mail or mail-in database. Finally, note that this database for storing backup copies of the ID can be

custom-named each time. You can create an ID recovery DB for each certifier. Figure 3-4 illustrates the simple view of documents into which the encrypted backup copies of the ID are stored.

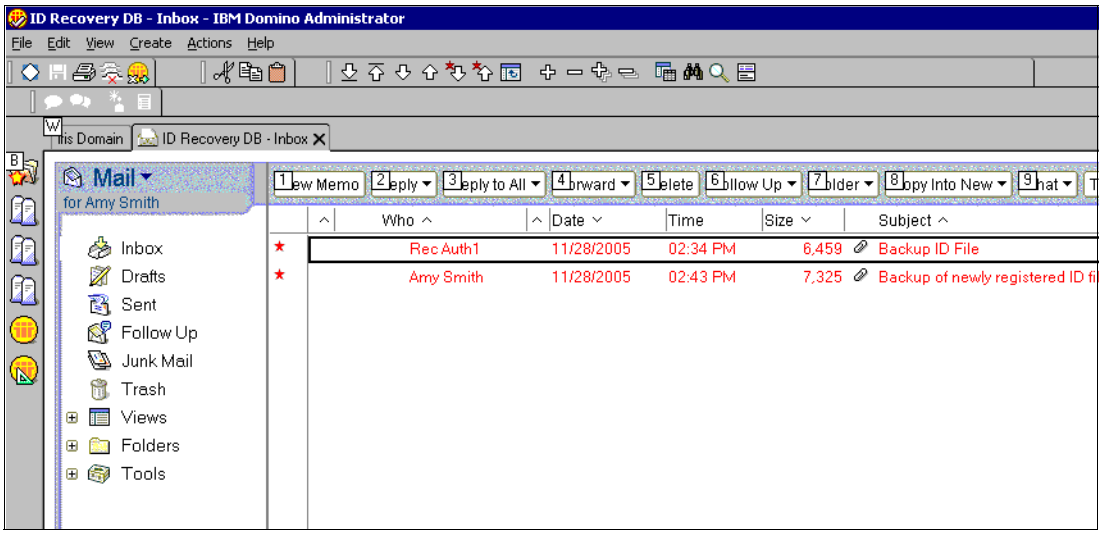


Figure 3-4 View of documents into which the encrypted backup copies of the ID are stored

Recovering an ID file for which the password has been forgotten is a bit easier. If the original ID file contains recovery information, administrators can recover the ID file, even if an encrypted backup ID file does not exist. This “recovery” information needed to recover an ID file is maintained both in the Domino certifier ID and in the user’s ID file.

The certifier ID file stores the following information:

- The number of Notes administrators required to unlock an ID file
- The names of Notes administrators who are allowed to recover IDs
- The address of the mail or mail-in database where users send an encrypted backup copy of their ID files

It is possible for the Notes administrator to set up ID recovery for user IDs at any time. If this is done before users are registered, ID recovery information is automatically added to user IDs the first time that users authenticate with their home servers. If ID recovery information is set up after Notes users have been registered, recovery information is automatically added to the user IDs the next time users authenticate with their home servers.

Tip: It is a best practice to designate several administrators who will act as a group to recover IDs and passwords. Although a single administrator can be designated to manage ID recovery, consider having two or more administrators work together to recover ID files. Designating a group of administrators helps to prevent a breach of security by one administrator having access to all ID files. When a group of administrators is designated, it is possible to specify that only a subset of them be present during the actual ID recovery. For example, if five administrators are designated for ID recovery but it is only required that three administrators be present to unlock the ID file, any three of the five can unlock the ID file. Designating a group of administrators and requiring only a subset also prevents problems that occur if one administrator is unavailable or leaves the company.

Before ID files can be recovered, an administrator who has access to the certifier ID file must specify recovery information, and the ID files themselves must be made recoverable. There are three ways to do this:

- At registration, administrators create the ID file with a certifier ID that contains recovery information.
- Administrators export recovery information from the certifier ID file and have the user accept it.
- (Only for Domino 6 servers and later) Administrators change recovery information using a Domino 7 Administrator client. Subsequently, recovery information is added automatically to users' Notes IDs when users authenticate to their home server.

ID recovery and smartcards: If Notes users will be enabling smartcards to use with their Notes IDs, it is extremely important for Notes administrators to set up ID recovery information for these IDs before any Internet keys are pushed onto the smartcard. Otherwise, the ID file recovery process will not be able to restore those keys. Additionally, acquiring recovery information, through any means, makes any Internet keys that had been previously pushed to the Smartcard unrecoverable.

3.3 How ID recovery works

For each Notes administrator, the user's Notes ID file contains a recovery password that is randomly generated and encrypted with the administrator's public key. The password is unique for each administrator and user.

For example, Notes administrator Laurent Hoerni has a unique recovery password for user Noel Doyle and that password is stored in Noel's ID file. Similarly, Notes administrator Jean-Jacques Chambaz has a unique recovery password for user Daniel Coddron and that password is stored in Daniel's ID file.

3.3.1 Ability to determine recovery password strength

In Domino 7, it is now possible to select the number of characters, or password length, for recovery passwords, which helps determine password strength, or likelihood to be compromised.

A password length that is fewer than 16 is calculated using both alphanumeric characters and hexadecimals. Sixteen-character length passwords are generated using hexadecimals only.

While password strength is important, because a strong password is less likely to be compromised, so is usability. A long and complex password can be difficult to use, so administrators also have the ability to choose a shorter password length.

In addition, administrators can now configure a custom message to help walk users through ID recovery.

When users acquire a new public key, accept a name change, or accept or create a document encryption key, Domino *automatically* sends updated encrypted backup ID files to the centralized database. In the case of a server-based certificate authority, the recovery database will be updated after the user has connected to the server.

Note: Recertifying a user does not generate an encrypted copy of the ID file to be sent to the recovery database, because a user's Person document already contains the updated public key.

3.3.2 Moving to a different certifier ID

If a user has been renamed by or moved to a different certifier that contains recovery information that is older than that of the user's previous certifier, the new certifier's recovery information will not be accepted into the user's ID file. Before using the new certifier, its recovery information must be updated so that it is more recent than the previous certifier's recovery information. To do this, the administrator should modify the new certifier's recovery information in some way and save it. This updates the recovery information for that certifier with a new time stamp and ensures that users who are subsequently renamed with or moved to the updated certifier will have the correct recovery information propagated to their user IDs. The administrator can then undo the change, if desired.

To help prevent unauthorized users from recovering IDs without the authorized user's knowledge, ensure that password verification is enabled for users and servers. If password verification is enabled, the authorized user is aware of the change because the user cannot access servers using the legitimate ID. When the unauthorized user recovered the ID file, that user was forced to make a password change.

As an extra precaution, after recovering IDs, users should be asked to re-accept the recovery information and then change the public key on their ID files. Re-accepting recovery information changes recovery password information in the ID file. As of Domino 6, re-accepting recovery information happens automatically when the user accesses a database on the home server. Changing the public key changes the public and private keys stored in the ID file.

3.4 ID recovery logging

Beginning with Domino 7, important information about client ID recovery activities are automatically logged to the local LOG.NSF file so that this information is available to administrators for troubleshooting purposes.

The following ID recovery information is logged locally:

- Date and time when recovery information is accepted into the ID file
- Instances when recovery information is rejected or fails to be accepted in the ID file
- Events that require a new backup to be mailed to the ID recovery database
- E-mailing the recovery ID to the recovery database (successes and failures)

Figure 3-5 on page 37 illustrates some log information relevant to ID recovery.

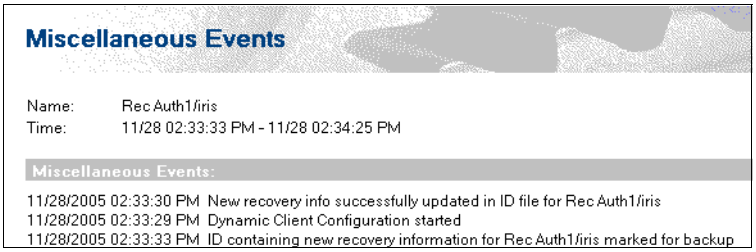


Figure 3-5 Logging for ID recovery

3.5 Process for configuring and setting up ID recovery

Now that the groundwork has been laid regarding ID recovery, we discuss how to set up and configure ID recovery.

Before users can recover their ID files, you must set up ID recovery. Perform these steps before anyone loses or corrupts an ID, ideally *before* you begin registering users:

1. From the Domino Administrator, click **Configuration**, and then click **Certification**.
2. Click **Edit Recovery Information**.
3. In the Choose a Certifier window, click **Server** and select the registration server name from the Domino Directory (only if the correct server name does not appear).
4. Choose the certifier for which you are creating recovery information:
   - If you are using a server-based certificate authority, click **Use the CA process** and select a certifier from the drop-down list. You must be a certificate authority (CA) administrator for the certifier in order to change ID recovery information.
   - If you are not using a server-based certificate authority, click **Supply certifier ID and password**. If the certifier ID path and file name does not appear, click **Certifier ID** and select the certifier ID file and enter the password.
5. Click **OK**. The Edit Master Recovery Authority List window opens.

   Figure 3-6 on page 38 shows the window for ID recovery setup in the R7 Administrator.

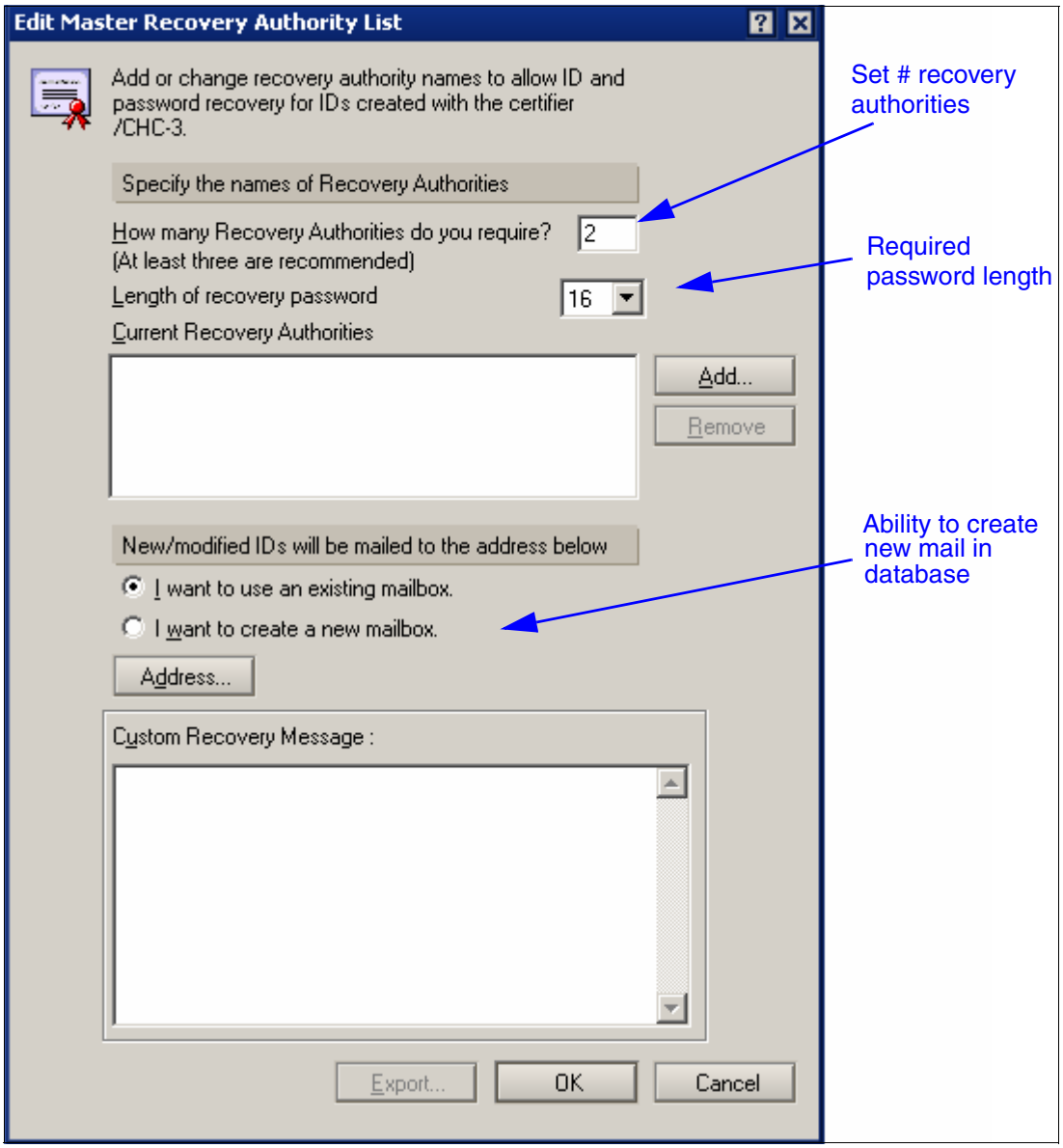


Figure 3-6 ID recovery setup in the R7 Administrator

6. Enter the number of recovery authorities that are required to recover an ID file. We recommend that you choose at least three.
7. Select the length of the recovery password from the drop-down list. The default is 16 characters.
8. Click **Add** and select the names of the administrators who are the designated recovery authorities.
9. Choose whether you want to use an existing mailbox for recovery information or create a new one:
   - If you have a mail or mail-in database already set up for recovery information, click **I want to use an existing mailbox**. Click **Address** and select the database from the Domino Directory.

– If you want to create a new database to store recovery information, click **I want to create a new mailbox**. In the Create New Mailbox dialog box, enter the name of the server on which the database is to be created and the database title. You can use the file name that is created from the database title, or you can create a new one.

10. In the Custom Recovery Message field, type a customized message for the Enter passwords dialog box that opens during the ID recovery process. For example, you might want to specify help desk contact information. The message length is limited to 512 characters.

    Note: Whenever you make changes in this dialog box, the Export button is disabled. You cannot export recovery information until you save the new or updated information.

11. Click **OK**.
12. If you are using a server-based certificate authority, at the server console, type:

    ```
    load ca
    ```

    This starts the CA process with the new recovery information, or refreshes it if it is already running. Then type the following command to process the request to add recovery information to the certifier:

    ```
    tell adminp process all
    ```

13. In the mail-in database access control list (ACL), set the Default access to No access and give administrators Reader access.

    Note: If you have created additional O-level Notes certifiers, be sure to cross-certify them with the initial Notes certifier prior to setting up recovery information.

Preparing IDs for recovery

After you specify recovery information in the certifier ID, when you register users, the user IDs automatically contain recovery information. However, if you specified recovery information after generating user IDs, users must update their user IDs with recovery information supplied by the administrator. Updating IDs with recovery information automatically sends an encrypted backup of the user ID to the centralized mail or mail-in database.

There are two ways that users can update their user IDs with recovery information:

- (Only for Domino 6 servers and later) Users authenticate to their home server after an administrator has added recovery information to the certifier. The recovery information is automatically added to their Notes ID.
- The administrator sends recovery information to users to incorporate into their user IDs. You must complete these steps before a user loses or damages an ID or forgets a password.

In Domino 7, users can determine whether recovery information is present in their user ID by seeing whether the “Mail Recovery ID” button in the User Security dialog box is active. They can then click the button to send an encrypted backup of the user ID to the centralized mail or mail-in database.

3.6 To send recovery information to the user

The administrator completes these steps:

1. From the Domino Administrator, go to the Configuration tab, and then click **Certification**.

2. Click **Edit Recovery Information**.
3. In the Choose a Certifier dialog box, if the correct server name does not appear, click **Server** and select the registration server name from the Domino Directory.
4. Choose the certifier for which you are creating recovery information:
   - If you are using a server-based certificate authority, click **Use the CA process** and select a certifier from the drop-down list.
   - If you are not using a server-based certificate authority, click **Supply certifier ID and password**. If the certifier ID path and file name do not appear, click **Certifier ID** and select the certifier ID file and enter the password.
5. Click **Export**, and then enter the certifier ID's password.
6. Complete the fields shown in Table 3-1, and then click **Send**.

Table 3-1 Fields for the administrator to complete

Field	Enter
To	Names of users and groups whose ID files you want to back up.
CC	Names of users and groups to whom you want to send a copy of the message.
Subject	Information for users and groups that will appear in the Subject field of the message. If this field is blank, Notes uses the following text: `New ID file recovery information is attached. Please add it to your ID file by using the Actions menu "Accept Recovery Information" option.`
Memo	Information for users and groups that will appear in the body field of the message. Domino automatically attaches the encrypted backup file information to the message; you do not need to specify it in this field.

To accept recovery information in the ID file

The user completes these steps:

1. After the administrator sends the recovery information, open the message in your mail database.
2. Select **Actions** → **Accept Recovery Information**, and then enter your password.
3. Complete the fields shown in Table 3-2, and then click **Send**.

Table 3-2 Fields for the user to complete

Field	Enter
To	Name of the mail or mail-in database that will store the backup copy of your ID. Domino enters the name of the database specified by your administrator.
CC	Names of users and groups to whom you want to send a copy of the message.
Subject	Information for administrators that will appear in the Subject field of the message. If this field is blank, Notes uses one of the following messages: ► Backup of newly changed recovery information for user name ► Backup of recent changes to ID file for user name
Memo	Information for administrators that will appear in the Body field of the message. Domino automatically attaches the backup of the ID file to the message; you do not need to specify it in this field.

Domino automatically sends the encrypted backup ID file to the centralized mail or mail-in database specified by the administrator.

Note: You can store multiple copies of the ID file in the centralized mail or mail-in database. Domino creates a new document every time an ID file is backed up. When attempting to recover an ID file, use the most recent backup. If this fails, use the older versions.

Tip: Use the NOTES.INI variable `ID_Recovery_Suppress_Recovery_Msg` to suppress the creation of the recovery memo if you want to suppress the standard message that appears on the recovery e-mail and replace it with a custom message.

3.7 Process for recovering an ID

If a user loses or damages an ID file or forgets a password, the user can work with administrators to recover the ID file from backup. If the user does not have access to her user ID file, she can obtain an encrypted backup of the user ID.

To recover an ID, users and administrators perform the following steps:

1. The user contacts the designated administrator to obtain the administrator's recovery password.
2. The administrator obtains the recovery password by decrypting the recovery password stored in the user's ID file using the administrator's private key.
3. The administrator then gives the recovery password to the user.
4. The user repeats steps 1 through 3 until the minimum number of administrators to unlock the ID file is reached.
5. The user starts Notes and clicks **OK** in the password dialog box instead of entering a password.
6. In the “Wrong password” dialog box, the user clicks **Recover password**.
7. The user selects the user ID file to recover in the Choose ID File to Recover dialog box.
8. The user enters the password provided by the administrator or administrators in the Enter Passwords dialog box, and repeats this until all of the passwords have been entered.
9. After the file is unlocked, the user is then prompted to enter a new password for their user ID. If the user does not provide a new password, the user needs to recover the user ID again.
10. Users should replace all backups and copies of their user ID files with the newly recovered user ID file.

Tip: The same ID file can be recovered again using the same recovery passwords. However, you should urge users to refresh the recovery information and create a new backup by re-accepting the recovery information after they recover their ID files.

To obtain the ID file recovery password

For security reasons, we recommend that administrators complete these steps from their own workstations, rather than from the same workstation. Using separate workstations prevents an unauthorized user from using a program to capture the keystrokes that the administrators enter on the same workstation. If an unauthorized user obtains an administrator's ID file and password, the unauthorized user can obtain the administrator's recovery password for all ID files. Therefore, you must protect the administrator's ID file and require that multiple administrators work together to recover any given user ID file.

Perform the following steps:

1. Detach the encrypted backup of the user's ID file from the mail or mail-in database to the local hard drive.
2. If the user's ID file is damaged, send a copy of the ID file from the centralized mail or mail-in database to the user.
3. From the Domino Administrator, go to the Configuration tab, and select **Certification** → **Extract Recovery Password**.
4. Enter the password to the administrator's ID file.
5. Specify the ID file you want to recover. This is the same ID you detached in step 1.
6. Note the recovery password. Give the user the recovery password that is displayed.

Note: The Extract Recovery and Recover ID File dialog boxes now display time stamp information for the recovery information contained in the copy of the ID file being recovered. Each time recovery information is generated or regenerated for an ID file, the recovery passwords all change. Occasionally, the recovery cookie acquired by an administrator cannot unlock a user's ID file; the recovery information had been regenerated at some point, and administrator is using a copy of the ID file that has a different set of recovery information. In situations like this, administrators can check the time stamp information displayed in these dialog boxes to see if they are trying to recover an ID file with outdated recovery information.

3.8 Changing administrator information for ID recovery

If an administrator leaves an organization or changes job responsibilities within an organization, you need to update the administration recovery information used to recover user ID files and then send the new information to users to add to their ID files. For Notes and Domino 6.0 or later users, the updated recovery information is automatically accepted into the ID file the next time the users authenticate with their home servers by accessing a database on the server.

To add or delete administrators

An administrator with access to the certifier ID completes these steps:

1. From the Domino Administrator, go to the Configuration tab, and then click **Certification**.
2. Click **Edit Recovery Information**.
3. In the Choose a Certifier dialog box, if the correct server name does not appear, click **Server** and select the registration server name from the Domino Directory.
4. Choose the certifier for which you are creating recovery information:
   - If you are using a server-based CA, click **Use the CA process** and select a certifier from the drop-down list.
   - If you are not using a server-based CA, click **Supply certifier ID and password**. If the certifier ID path and file name does not appear, click **Certifier ID** and select the certifier ID file and enter the password.
5. Do one of the following actions:
   - To delete an administrator, highlight the administrator's name, and then click **Remove**.
   - To add new administrators, click **Add** and then select the names of administrators who are authorized to recover ID files.
6. (Optional) Change the number of administrators required to unlock an ID.

7. When you finish adding or deleting names, click **OK**.

3.9 Summary checklists

The following section provides high-level checklists for the process of both setting up ID recovery and for recovering an ID.

3.9.1 Initial setup

For the initial setup, perform the following steps:

1. Decide which Domino server will be the CA server within your overall organization. The machine should be very reliable and physically secure.
2. On the CA server, edit the NOTES.INI file and add the `ca` task to the line `ServerTasks=`. Restart the server to start this task.
3. Create a mail-in database for use by the ID recovery process. The name of the mail-in can be ID Recovery and the location of the mail file should be on a trusted mail server (which can also be the CA server).

Creating and migrating the top-level certifier

Perform the following steps:

1. Create the top-level certifier by selecting **Domino Administrato**r → **Configuration** → **Registration** → **Organization**. (If you already have a top-level certifier, use that one. Do not create another.)
2. Migrate the top-level certifier to the CA process by selecting **Domino Administrator** → **Configuration** → **Certification** → **Migrate Certifier**. Perform the following steps:
   a. Select the top-level certifier ID file.
   b. Go to the Basics tab of the migration wizard, and enter the following information:
      - Set the name of your CA server.
      - Leave the name of the Internet Certifier List (ICL) file as is.
      - Encrypt the certifier ID with the **Server ID**.
      - Do not require a password to activate.
      - Set the names of the certifier authority administrators (CAAs) and registration authorities (RAs) as wanted.

        CAAs own the certifier and can make any change to it. RAs can use the certifier to create users, servers, and OUs with it. (Note that in R6 the CAA column is mislabeled as CA.)

        Because this is the top-level certifier, we suggest maintaining a fairly short list of people who are both CAAs and RAs.
   c. Go to the Certificates tab.

      We recommend leaving all of these defaults as is, except possibly the first entry in the first column. This specifies the default expiration for end-user IDs created with this certifier, which is currently 24 months. Some organizations set this to 12 months.

Figure 3-7 on page 44 shows the certifier migration dialog box from the R7 Administrator.

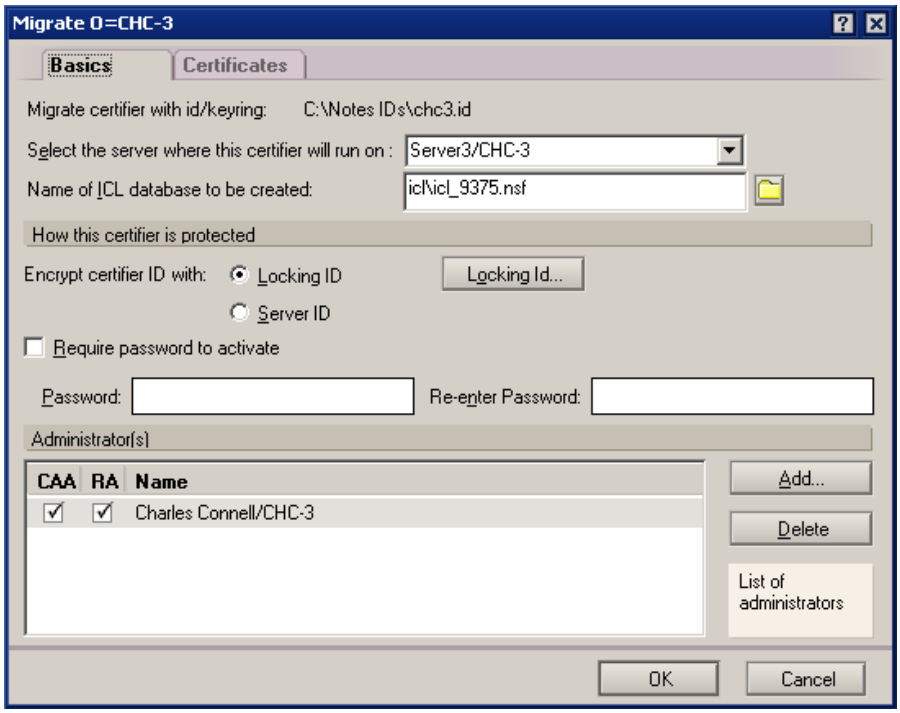


Figure 3-7 Certifier migration dialog box

3.9.2 Modifying the top-level certifier

You must be a CAA of the top-level certifier to perform these operations. You must also have Editor access to the Domino Directory (NAMES.NSF) for this Notes domain.

1. From the Domino Administrator, select **Configuration** → **Certification** → **Modifier Certifier**.
2. Set the CA server.
3. Select the certifier from the Domino Directory (instead of an ICL database).
4. Choose the certifier to modify.
5. Click **OK** to open the certifier.
6. Make the changes you want.
7. Click **OK** to save the changes.

Creating and migrating organization unit certifiers

Perform the following steps:

1. To create new organization-level certifiers, select **Domino Administrator** → **Configuration** → **Registration** → **Organizational Unit**. (This only applies to new OUs that you do not already have. For existing OUs, just migrate them to the CA.) Follow these steps:
   a. Select the CA server.
   b. Select **Use the CA process**.
   c. From the list of certifiers in the CA, choose the top-level certifier that will create the new OU.
   d. Enter the standard information about the new OU certifier, and then click **Register**.

2. Migrate the OU certifier to the CA process by selecting **Domino Administrator** → **Configuration** → **Certification** → **Migrate Certifier**. Follow these steps:
   a. Select the OU certifier ID file.
   b. Go to the Basics tab of the migration wizard. Provide the following information:
      - Set the name of your CA server.
      - Leave the name of the ICL file as is.
      - Encrypt the certifier ID with the **Server ID**.
      - Do not require a password to activate.
      - Set the names of the certifier authority administrators (CAAs) and registration authorities (RAs) as wanted.

        CAAs own the certifier and can make any change to it. RAs can use the certifier to create users with it. (Note that in R6 the CAA column is mislabeled as CA.)

        Because this is an organization-level certifier, we suggest a short list of CAAs and a longer list of RAs who work within that organization.
   c. Go to the Certificates tab.

      We recommend leaving all of these defaults as is, except possibly the first entry in the first column. This specifies the default expiration for end-user IDs created with this certifier, which is currently 24 months. Some organizations set this to 12 months.

Note: You must give RAs the following additional access rights:

- NAMES.NSF: Author access, Create Document right, UserCreator, and UserModifier roles
- CERTLOG.NSF: Author access, Create Document right

3.9.3 Modifying organization unit certifiers

You must be a CAA of the OU to perform these operations. You must also have Editor access to the Domino Directory (NAMES.NSF) for this Notes domain. Perform the following steps:

1. Select **Domino Administrator** → **Configuration** → **Certification** → **Modifier Certifier**.
2. Set the CA server.
3. Select the certifier from the Domino Directory (instead of an ICL database).
4. Choose the certifier to modify.
5. Click **OK** to open the certifier.
6. Make changes as you want.
7. Click **OK** to save the changes.

3.9.4 Setting up ID recovery

Perform the following steps:

1. Select **Domino Administrator** → **Configuration** → **Certification** → **Edit Recovery Information**.
2. Select the CA server.
3. Select **Use the CA process**.
4. From the list of certifiers in the CA, choose the certifier you want to edit.

5. Enter the ID recovery information.
6. Configure the cookie length. For a cookie length greater than 16 characters, you require Domino Administrator 7 or later.
7. Specify the ID recovery mail-in database that you previously created.
8. Click **OK** to save the recovery information.

After modifying and saving the recovery information, it is automatically copied to user ID files when users access their home servers. Updated copies of user ID files (with the new recovery information) are automatically sent to the ID recovery mail-in database as part of this process.

3.9.5 Modifying ID recovery information

Modifying ID recovery information is the same as setting it up for the first time. Users will receive the updated information in the same way, when they access their home servers.

3.9.6 Recovering a forgotten password: User actions

A user can verify the availability of recovery information by selecting **File** → **Security** → **User Security** → **Basics**. The Mail Recovery ID button, if enabled, indicates that the ID has recovery information. Click it to send an encrypted backup to the recovery database.

3.9.7 Recovering a lost or corrupted ID file

Recovering a lost or corrupted ID file is exactly the same as recovering the password for an existing ID file, with one addition. Before the process can begin, the user asks the administrator to send him or her a copy of the backup ID file.

Because the user is locked out of their Notes workstation, the administrator cannot simply send the ID to the user by e-mail. Instead, the user must receive the backup ID file by walking to the administrator's office (where this is possible), through a coworker's e-mail account, or by physical mail on a diskette or CD.

After the user has the backup ID file, recovery can proceed just as though the user had forgotten the password for the ID.

4

Smartcards

Smartcards, a category including, for our purposes, both the credit card form factor (read by a reader) and cryptographic tokens (plugged directly into USB port), represent a significant, cross-application advance in user security. Designed to be under the physical control of the owner at all times, smartcards bolster the foundational rule that security is enhanced when the owner brings what she uniquely *has* (a cryptographic token, an ID file) together with what she uniquely *knows* (a passphrase). Industry support and marketplace acceptance have reached a critical mass where the promise of smartcard technology, a decade in the making, is being realized, not just recognized. Adoption by security-conscious organizations is decidedly on the rise.

Note: Discussion here is up-to-date as of the initial release of Lotus Notes 7 (and codestreams 6.5.4 and 6.0.5). Notes smartcard support will continue to develop over time, perhaps rapidly. We encourage you to check the *Release Notes* and online documentation of later versions (and simply test directly) to determine if any noted limitations continue to apply. To access the *Release Notes*, select the **Release Notes** link in the Documentation Links section at:

`http://www.lotus.com/doc`

4.1 Why smartcards?

Other factors held constant, the more a system ensures that a user (or application, thus "an owner") is "personally" presenting that which only that user should possess, the stronger the authentication and the greater the security of a system. Without smartcards, if an attacker comes to control a copy of a target's Notes ID file, only the password encrypting the ID stands between the attacker and system compromise. An attacker can overcome the password by brute force or surreptitious social means. In addition, a breach might go undetected and even unsuspected.

Simple use of a smartcard can alleviate these dangers and threats significantly. Compromise becomes much more difficult because a physical card or token must be stolen as well, and the token typically does not leave the owner's side (for example, it is not stored on a networked computer's hard-drive over the weekend). Further, if the smartcard is stolen, it typically locks out after a few failed PIN[1] challenges (resettable only through administrator intervention, if that). Its disappearance will tend to be noticed quickly because dependent systems will halt or break as soon as the protected credentials are needed (for example at the next logon attempt or next private cryptographic event, that is an attempt to authenticate, decrypt, or sign). Therefore, if all extant copies of a particular Notes ID are secured by a smartcard, the risk of credential compromise is very unlikely if only the file itself is breached.

Embedded in a smartcard device are both a microprocessing chip where cryptographic operations execute solely "in hardware," and an amount of secure storage. *In hardware* is meant in opposition to "in the memory space of the host computer," space usually accessible in some way by any other code executing on the host. The secure storage holds private keys or other cryptographic objects that the smartcard typically will not allow to be copied by users or third-party applications, if ever at all. (Other stored items might be designated public and so permitted by the smartcard to be copied.)

4.2 Smartcard installation

The smartcard implementation landscape is unfortunately unsettled: competing vendor implementations, competing standards, disparate and conflicting implementations of ostensible standards, and so on. Fortunately, the RSA PKCS #11, Cryptographic Token Interface Standard, has gained significant traction over the last few years, meaning cross-vendor applications and benefits are now being realized and implemented widely. Lotus Notes endorses and participates in PKCS #11.

While the number of smartcard products tested with Notes directly by Lotus is limited, any smartcard that provides a PKCS #11 interface and supports a minimum set of functionality charted by version 2.01+ of the standard[2] should succeed. Limited functionality might be possible with a smartcard providing a version 2.0 interface, but we do not recommend such implementations. Smartcards with a PKCS #11 "Protected Authentication Path" (an on-board interface for input of PIN or equivalent, for example, a fingerprint reader) are supported as of Notes 6.0.1. X.509 certificates pre-loaded onto a smartcard fulfilling certain standard functionality[3] can be imported into a user's ID file as of Notes 6.0.2. As of Notes 7, the smartcard to which Notes binds need no longer be in first position on the system (the first "slot").

---

[1] Personal identification number, although it is often not limited to numerals

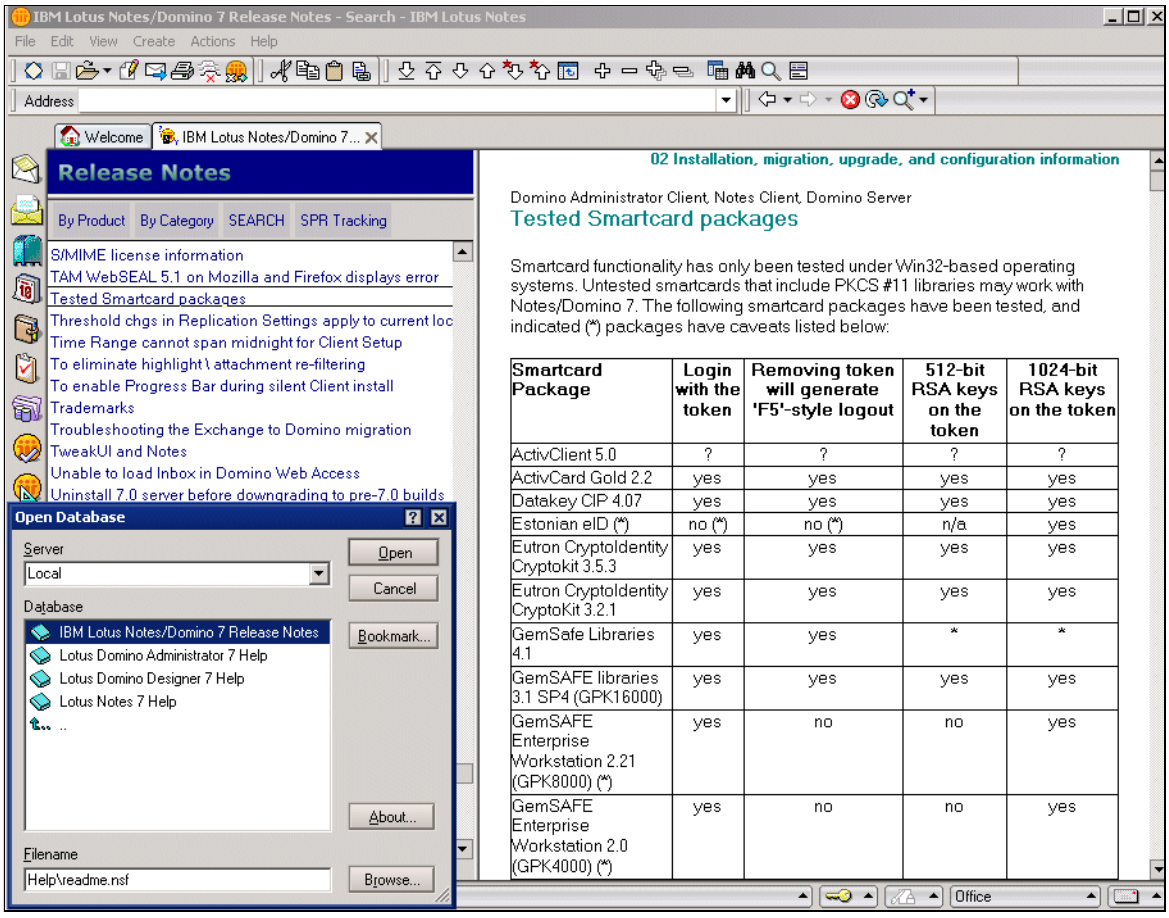
[2] Specifically, "Large Applications Profile" described in the standard's "Conformance Profile Specification" document, available at: http://www.rsasecurity.com/rsalabs/node.asp?id=2133

[3] Specifically, the "RSA Asymmetric Client Signing Profile" of the "Conformance Profile Specification

Although improving, smartcard installation and function remain difficult. We strongly recommend that organizations interested in achieving the benefits of smartcards measure twice, with *planned* system testing and a representative pilot, before attempting a wide deployment. Use this same process when upgrading a smartcard's driver.

Tip: Test a smartcard's function in isolation and in stages. Faulty installation can lead to instability and severe problems on a system because credential management and manipulation is involved. One difficulty we experienced, for example, is full impairment of a system's logon function. The installer might show a Microsoft Windows® logon, Graphical Identification and Authentication (GINA, a Microsoft acronym), option, for example, perhaps selected by default, that might be buggy or outdated. Therefore, we advise against activating non-core features at the outset. After a smartcard has proven itself against Notes, you can then layer on and test extra features.

Because the implementation landscape is unsettled, we do not include a Notes compatibility matrix. We advice architects and administrators to refer first to the *Release Notes* shipped with the Notes client version serving their user population. Figure 4-1 shows an example *Release Notes* version and its location.



Smartcard Package	Login with the token	Removing token will generate 'F5'-style logout	512-bit RSA keys on the token	1024-bit RSA keys on the token
ActivClient 5.0	?	?	?	?
ActivCard Gold 2.2	yes	yes	yes	yes
Datakey CIP 4.07	yes	yes	yes	yes
Estonian eID (*)	no (*)	no (*)	n/a	yes
Eutron CryptoIdentity Cryptokit 3.5.3	yes	yes	yes	yes
Eutron CryptoIdentity CryptoKit 3.2.1	yes	yes	yes	yes
GemSafe Libraries 4.1	yes	yes	*	*
GemSAFE libraries 3.1 SP4 (GPK16000)	yes	yes	yes	yes
GemSAFE Enterprise Workstation 2.21 (GPK8000) (*)	yes	no	no	yes
GemSAFE Enterprise Workstation 2.0 (GPK4000) (*)	yes	no	no	yes

Figure 4-1 Smartcard support matrix example in Notes client Release Notes database

Tip: To access the *Release Notes*, select the **Release Notes** link in the Documentation Links section at:

`http://www.lotus.com/doc`

To enable a smartcard on a system, install the vendor's drivers, including any PKCS #11 interface if optional. Further, if you do not use a smartcard preconfigured for immediate use, install any tooling needed to configure the smartcard for regular use, for example, PIN initialization, perhaps needed X.509 credential import.

To register the smartcard with the Notes client, in the User Security dialog box, bind it by selecting **Your Identity** → **Your Smartcard**, as shown in Figure 4-2 (access the User Security dialog box, in the Notes client, select **File** → **Security** → **User Security**). In order to function here, the smartcard must be plugged into the system and the binary library file that provides the required PKCS #11 interface must be found (in Windows, a DLL[4]). Typically, the library is provided by the smartcard vendor and placed in the system's binaries directory (in Windows, system32). Specification can occur either through a special dialog box prompt that the Notes client opens when the user navigates to the panel, or through the Configuration Details button on that panel. After the library is specified correctly, the smartcard's identifying and capability information are queried by Notes and displayed in appropriate fields (for example, those circled in Figure 4-2). Success here suggests that further interaction between Notes and the smartcard can proceed.

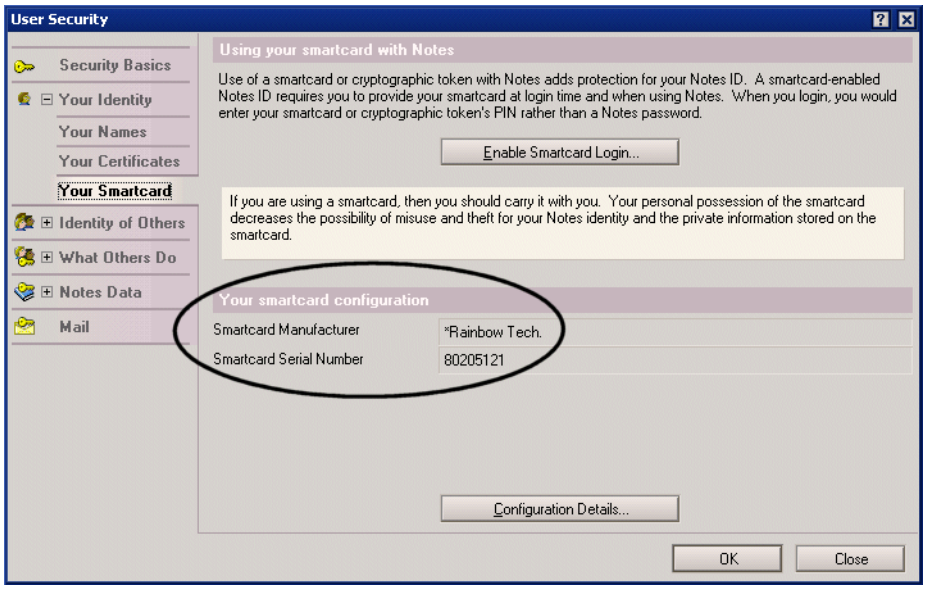


Figure 4-2 Smartcard configuration panel on the User Security dialog box

Note: The smartcard functionality in the User Security dialog box on a Macintosh Notes client might differ from that shown on a Windows client. Lotus has tested smartcards only on Windows Notes clients.

4.3 Notes client smartcard functionalities

Because smartcard support was introduced initially with Notes 6 and documentation is thin beyond the step-by-step how-to in Notes online help, we provide a general discussion of smartcard functionalities in Notes. We also highlight the enhancements introduced with Notes 7.

[4] Notes' smartcard support extends to the Macintosh client. We tested smartcards only on Windows Notes clients, however.

Notes smartcard functionality extends to three areas:

- Secure the Notes ID file itself, such that the smartcard must be presented to unlock the ID so that operations requiring a private key can proceed.
- Use a private X.509 key stored on the smartcard in S/MIME data-security operations (signing and decrypting) and in SSL client authentication. Facilitating this, Notes supports X.509 key exchange between the smartcard and an ID file.
- With large-key Notes IDs (1024+ bit), use a private Notes key stored on the smartcard in data-security operations (Notes signing and decrypting) and in client/server authentication.

4.3.1 Smartcard-securing a Notes ID

A Notes ID is secured by encrypting the private portion of the file with a user-supplied secret. Traditionally, the secret has been a user-known passphrase. Now, the secret can be held and secured on-board a smartcard. The secret can take one of two forms:

- A special private data object, the object representing a very complex, untypable passphrase
- Beginning with Notes 7, a private X.509 key that has been linked to the user's ID file

When an ID is secured in one of these ways, it is said to be "smartcard-enabled" (smartcard-secured). After smartcard-secured, unlocking the ID requires the presentation and unlocking of the smartcard. In keeping, a smartcard-secured Notes client treats removal of the smartcard as a logout action (similar to pressing the F5 key).

In keeping with smartcard conventions, Notes does not permit an ID's smartcard binding to be reversed to traditional passphrase unlocking, because this violates the principle that smartcard presentation should be permanently required to unlock the resources it has been assigned to guard, a flexibility restriction that strengthens the security solution in regular use.

Note: As an administrative fallback, ID recovery can be used to revert an ID file to make it traditionally passphrase secured. For more information about this, see 4.5, "Considerations and caveats" on page 58.

Placement of a special, private passphrase object

Notes 6 introduced the initial method used to smartcard-secure a Notes ID. In the User Security dialog box, select **Your Identity** → **Your Smartcard** and then click the **Enable Smartcard Login** button, as shown in Figure 4-2 on page 50. This creates and stores a 64-byte random passphrase on the smartcard. The passphrase is roughly equivalent in strength to a 512-bit symmetric key (extremely strong), salted, and adjusted to ensure that it is untypable (that is, bytes introduced that do not map to a character) and unusable with existing Notes C API functions (embedded null characters thwart that approach).

We encountered one minor limitation with this passphrase-object method of ID security with a Rainbow iKey 2032. The scenario involved a user with two separate copies of her ID file. If one copy was secured by passphrase object, an attempt to secure the other copy with the smartcard failed with an error. (However, testing showed that IDs of multiple users can be secured by a single smartcard using the passphrase-object method.) If a smartcard user needs to roam freely among a set of Notes installations and the passphrase-object approach is required, secure one ID file and copy it through the file system to replace the others. Therefore, only one passhprase object is in use, and the one smartcard unlocks all the ID copies because they are identical. The user gets the benefits of increased security (the ID files are useless without the smartcard) and easier usability (password/PIN guaranteed to

remain consistent across the installations). Note, in any case, that the cited limitation does not apply if ID files are secured by a smartcard-based private key, as discussed in the following section.

Association with a resident, linked X.509 private key (Notes 7)

Notes 6 introduced the ability to copy certain appropriate X.509 key material between a smartcard and an ID file (see 4.3.2, “Smartcard X.509 key linking” on page 53) and to use smartcard-resident X.509 private keys, linking the ID to such keys for normal X.509 security functions (S/MIME data security and SSL client authentication). Notes 7 improves on this start in an important way: allowing a private X.509 key stored on a smartcard to secure the Notes ID itself, instead of the unconventional special passphrase object. Powerful implications flow from this. First, we describe how to secure an ID with an X.509 key:

1. If the X.509 certificate and key pair are not present in the ID file or on the smartcard, they must first be imported to one of them. If importing to the smartcard, you typically use the smartcard vendor’s tooling and import a PKCS #12 encoded file holding the X.509 payload. By contrast, you can import to the Notes ID file by any means supported by Notes.

   Note: For example, this can be accomplished in the following ways:

   - Automatically upon Domino server authentication (carrying out an administrator’s prior-issued instruction)
   - Importing the certificate by clicking the **Get Certificates** button in the User Security dialog box, as shown in Figure 4-3

2. The available X.509 key pair must be bound to the side still ignorant of its existence (target smartcard or Notes ID). This occurs again through the User Security dialog box, as described in 4.3.2, “Smartcard X.509 key linking” on page 53.
3. Finally, the user must select **Other Actions** → **Lock ID File with Key on Smartcard**, as shown in Figure 4-3 on page 53. This menu item is enabled after completing steps 1 and 2 and the user has highlighted the target personal X.509 certificate. To access the certificate, in the User Security dialog box, select **Your Identity** → **Your Certificates**. Then select **All Internet Certificates**, **Your Internet Certificates**, or **All Certificates** from the drop-down list box accessed.

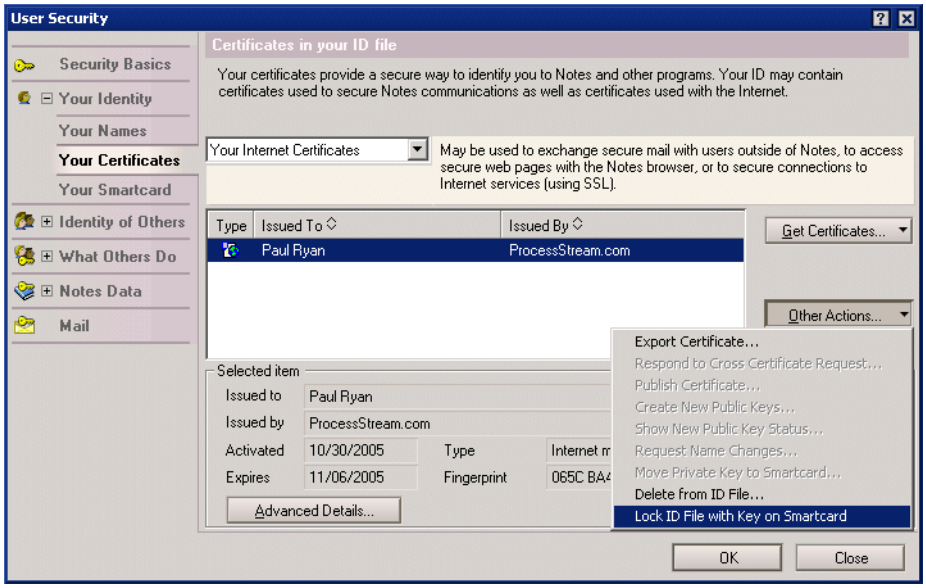


Figure 4-3 Lock ID File with Key on Smartcard

If the preceding steps begin with the personal X.509 certificate resident on the smartcard (before being linked to the ID and used to secure it), this provides more usage flexibility:

- Most importantly, read-only smartcards are supported (neither the special password object required in Notes 6 need be written, nor an X.509 private key).
- It follows, therefore, that separate copies of an ID file can be independently secured with the same physical smartcard, useful for roaming users. Indeed, even separate IDs can share a physical smartcard, although the utility of that scenario seems strained (a developer switching among test-user IDs, perhaps).

Interestingly, whichever way an ID is secured by an X.509 key, if the smartcard is lost or broken, a replacement can be generated if a copy of the full personal certificate (private key included) has been archived (a PKCS #12 file held in secure storage, for example). For regeneration to succeed, the PKCS #11 CKA_ID metadata property of the private key on the smartcard (and associated certificate and public key if present) must match the value it had on the original smartcard; Notes uses this property when locating keys on smartcards. Typically, the smartcard vendor's tooling generates this value consistently on import, so if enablement originally began that way (personal X.509 certificate preloaded on the smartcard), no further action should be required. For other scenarios, the CKA_ID property needs to be adjusted to match the original value expected by Notes, and some vendors' tooling supports this. Unfortunately, Notes does not yet display the CKA_ID value anywhere in the user interface, so if regeneration is of interest, an organization should keep a separate record of this non-private value.

4.3.2 Smartcard X.509 key linking

Smartcards customarily work with X.509 certificates and keys. Notes supports both the shifting of private X.509 keys onto a smartcard and the linking of certified keys preloaded on a smartcard to an ID file. In all cases, the result is that the X.509 private key resides only on the smartcard, where all private-key operations are conducted (S/MIME data-security and SSL client authentication).

Linking an ID to smartcard-resident certificates and keys

Often a user receives a smartcard already loaded with the required X.509 certificate and key pair. For example, a company that places an emphasis on security might give one to a consultant hired to undertake a sensitive project. The consultant can then import the X.509 certificate (public key included) into the Notes ID file, which also links the ID to the private key counterpart left on the smartcard. From then on, the consultant can send S/MIME e-mail signed by that key and decrypt mail encrypted to it, or use it against a Web site secured by SSL client authentication. To link this, from the User Security dialog box, select **Your Identity** → **Your Certificates**, select **Get Certificates**, and from the drop-down list, select **Import Internet Certificate from a Smartcard**, as shown in Figure 4-4. With Notes 7, this action does *not* require prior smartcard-securing of the ID file and no updates flow to the smartcard in the process, thereby supporting read-only smartcards.

Note that in this case of linking preloaded X.509 credentials to an ID, the presence of associated X.509 certificate objects are required, because Notes enumerates these when evaluating what on the smartcard can be imported or linked. Currently, this import step does not allow the user to choose which credentials should be linked; each credential that can be linked will be.

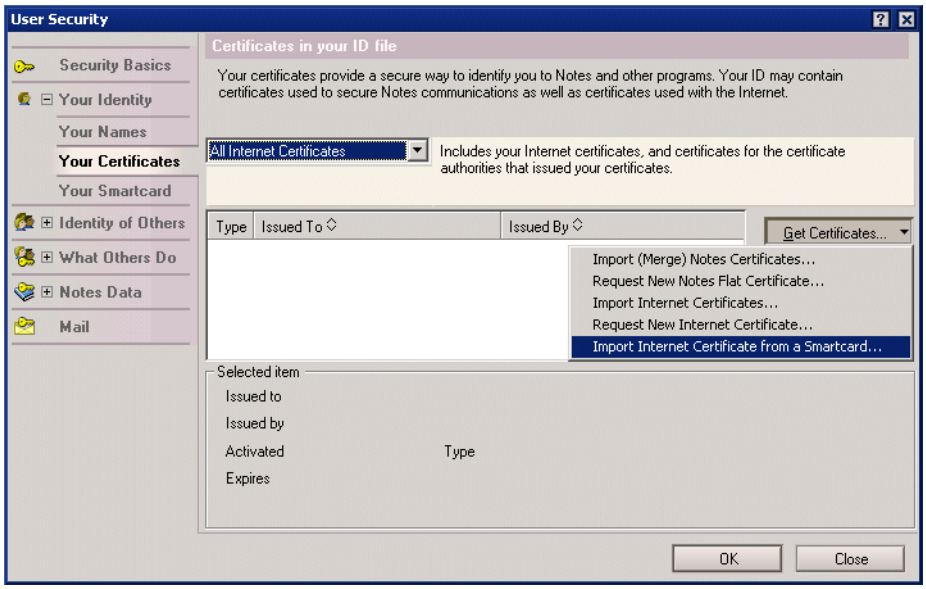


Figure 4-4 Import (and link to) an X.509 certificate on a smartcard

In order to observe an imported and bound certificate in the list box of the dialog box shown in Figure 4-4, the user might need to close and reopen the User Security dialog box. In any case, the result should look similar to that in Figure 4-5 on page 55. Note the small, gray, rectangular overlay at the top-left corner of the entry's Type icon (circled). This notation indicates that the private key associated with the certificate resides on a smartcard and therefore *not* in the ID file.

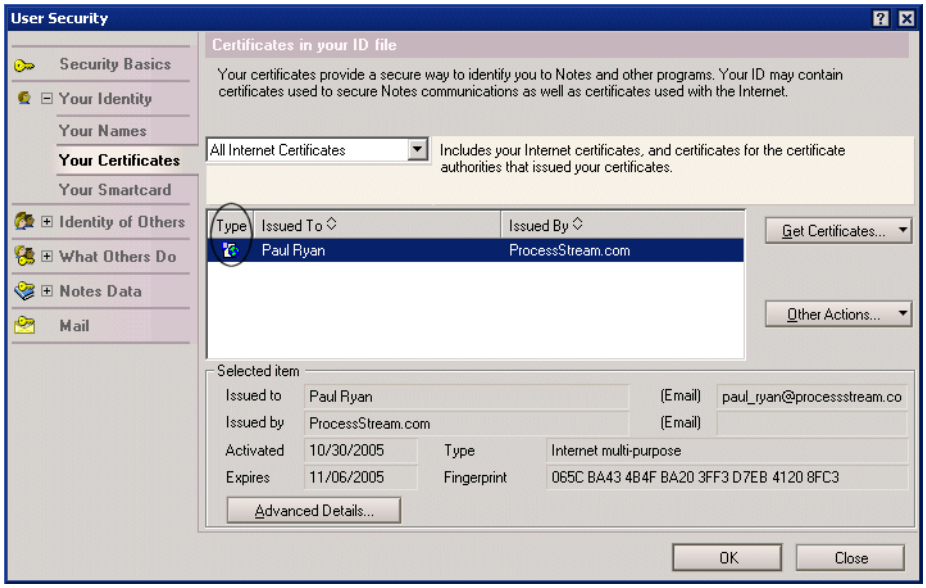


Figure 4-5 Personal X.509 certificate linked to a smartcard

Shift X.509 private key from a Notes ID to a smartcard

X.509 key linking can move in the opposite direction as well. Say a user receives an X.509 certificate and key pair into her ID file. This credential can be shifted to a smartcard, thus reaping the security benefits. To enable the shift, the ID must have already been secured by the target smartcard ("smartcard-enabled," as described in 4.3.1, "Smartcard-securing a Notes ID" on page 51). The user highlights the personal X.509 certificate whose private key should be shifted and selects **Other Actions** → **Move Private Key to Smartcard**, as shown in Figure 4-6. The private key is shifted to the smartcard and removed from the ID. In Notes 7, the associated public key is copied as well, facilitating key lookup in third-party situations where the smartcard's PIN is not required or available.

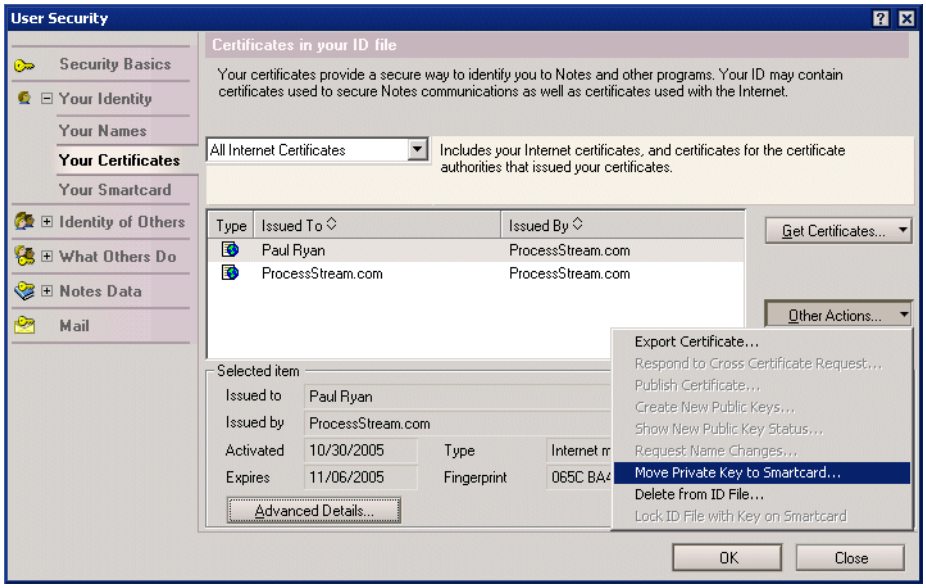


Figure 4-6 Move X.509 Private Key to Smartcard

If the ID file is enabled for Domino ID recovery, a copy of the X.509 private key is taken prior to its export to the smartcard. The copy is encrypted with the ID's separate, strong recovery public key and stored with the rest of the recovery information in the ID file. This action also marks the ID to be sent to the organization's secure ID recovery backup repository. For further information about how ID recovery intersects with smartcard-secured IDs, see 4.5, "Considerations and caveats" on page 58.

4.3.3 Shift a large Notes key to smartcard

Notes 7 introduced support for longer key lengths for the asymmetric key pair at the core of an ID (see Chapter 5, "Enhancements for longer keys in certificates and IDs" on page 61). These 1024+ bit Notes keys are stored and handled internally using more modern and widely adopted cryptographic standards than existed at Notes' inception (where one of the earliest PKIs was used and proved the longest lasting). One benefit of the modern handling provided with longer Notes keys is compatibility with PKCS #11-compliant devices, meaning that a large Notes key can be shifted to a smartcard using essentially the same means described in 4.3.2, "Smartcard X.509 key linking" on page 53 for X.509 keys, and with the same result: All private-key operations are conducted only by the smartcard's microprocessor. After a user's ID file is generated with or rolled over to a large Notes key, that user only needs to shift the key to the smartcard using the normal **Other Actions** → **Move Private Key to Smartcard**, as shown in Figure 4-7.

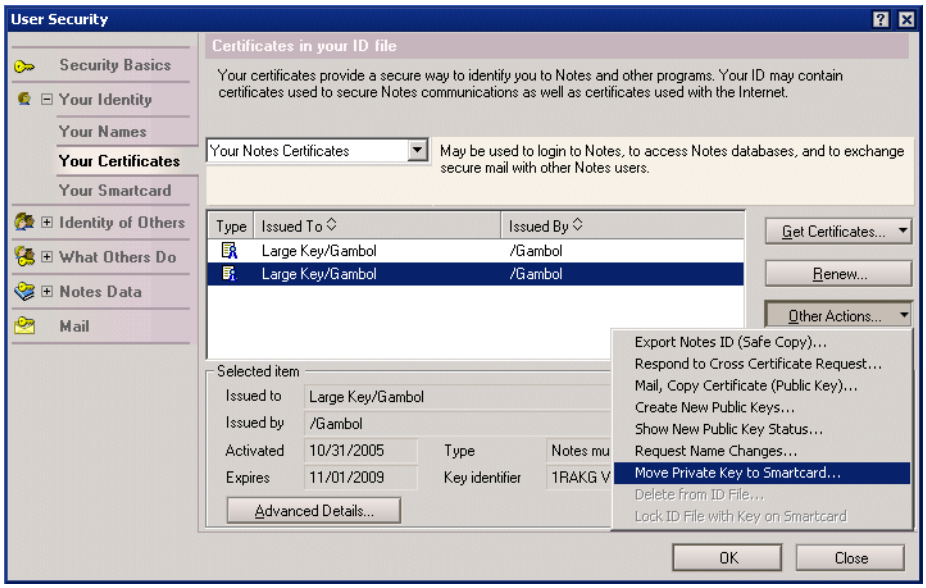


Figure 4-7 Move Private Key to Smartcard

Currently with Notes 7, the Type icon shown for a smartcard-linked Notes certificate does not include the tiny "on a smartcard" overlay, as shown in Figure 4-5 on page 55, with smartcard-linked X.509 certificates. Also, the initial Notes 7 client does not support smartcard-securing an ID file with a smartcard-resident Notes key in the manner described for X.509 keys in 4.3.1, "Smartcard-securing a Notes ID" on page 51; however, you can use a Notes C API program to accomplish this, as described in see 4.4.1, "Notes C API support" on page 57.

4.4 Extended functionalities

This section addresses the extended functionalities between Lotus Notes and smartcards: support lent by the Notes C API, securing of a Domino server ID with a smartcard, and support for cryptographic accelerator products offering a PKCS #11 interface.

4.4.1 Notes C API support

This section describes Notes C API support.

SECManipulateSC()

The all-in-one `Notes C API function SECManipulateSC()`, introduced in Notes 6.0.2, can programmatically perform all of the user-driven functionality described earlier, and more. Automation of the manipulation of smartcards through Notes IDs is therefore feasible.

For a thorough description of this function, refer to the reference database provided with the Notes/Domino 7 C API Toolkit. This toolkit is available thorough Lotus Developer Domain. Go the following site and select the **Toolkits and drivers** link from the Downloads and CDs section:

`http://www.lotus.com/ldd`

Beyond the functionalities revealed in the Notes client, with `SECManipulateSC()`, you can smartcard-secure a Domino key ring file (“Smartcard-securing the server” on page 57) and secure an ID with a large Notes key shifted to a smartcard. Implement this latter capability by using the following operation codes of the call:

- `SC_manip_FindAllKeys`: To determine the number of keys on the smartcard
- `SC_manip_GetMatchedCert`: In an enumeration loop, to locate the large Notes key (or any other RSA key) of interest and that key’s type determined by inspecting the `SC_MANIP_IMPORTABLE_CERT` structure output
- `SC_manip_LockIDWithKeyRO`: To secure the Notes ID with the specified smartcard-resident key

EM_GETPASSWORD

The Notes `EM_GETPASSWORD` Extension Manager hook is supported when a smartcard throws a PIN challenge. Therefore, using smartcards should not break solutions where this hook is used. The hook is particularly useful as a means to allow unattended system reboot, for example, although obviously this technique needs to be handled with care.

4.4.2 Domino server support

This section describes Domino server support.

Smartcard-securing the server

Domino server IDs can be secured with a smartcard similar to how user IDs are secured. However, PIN authentication will be required at server startup. Therefore, if automatic restart is a requirement, this approach is viable only if a Notes C API Extension Manager hook is formulated (see the previous section). Lotus Domino 7 Administrator Help discusses in detail how to involve a Notes client in securing the server’s ID file and notes the necessity of specifying the variable PKCS11_Library in the NOTES.INI file; see the “Physically securing the Domino server” topic.

Note that if the server ID includes a large Notes key pair (1024+ bit), we do not recommend shifting the private key portion to a conventional smartcard (a technique discussed in 4.3.3, “Shift a large Notes key to smartcard” on page 56), because this severely slows the many cryptographic operations needing the key (such as Notes client authentication and database decryption). We do not recommend shifting unless it be to a cryptographic accelerator.

Domino and cryptographic accelerators (SSL)

Lotus extended Notes and Domino 7 to support cryptographic accelerator products featuring a PKCS #11 interface. Cryptographic accelerators have conventionally been used to alleviate the severe performance penalty involved with SSL-encrypted network traffic. Optimized to execute cryptographic operations and loaded with parallel microprocessors, these products boost scalability when facing heavy loads of encrypted traffic. Historically, Domino provided no direct support for these devices. This changes with Domino 7.

Domino 7 brings support for multithreaded, multiconnection stream cryptography and for storing and accessing SSL private keys on PKCS #11-compliant devices. The Domino proprietary *.kyr key ring file format continues to be used in its SSL implementation, but Domino 7 now supports indirect access of the X.509 private key portion, similar to the private key indirection discussed earlier in regards to Notes IDs. However, a user interface has not been provided yet to effect the shift of a key ring file’s private key to PKCS #11 interfaced storage. Instead, for the time being, the Notes C API function `SECManipulateSC()` (see 4.4.1, “Notes C API support” on page 57) has been extended with the operation code `SC_manip_PushKyrKey`, meaning a short, simple Notes C API program must be coded to prepare a server’s key ring and ID files to engage in private key indirection.

As alluded to earlier, if the server’s ID is based on a large Notes key pair, the private key portion can be shifted to the accelerator’s PKCS #11 interfaced storage, thus benefiting server-based cryptographic operations involving that key (client authentication and database decryption). Port encryption does not use the asymmetric RSA private key directly, so the performance benefit does not extend to that context.

4.5 Considerations and caveats

In large part, smartcard-enablement of an ID *supplants* other password-centric security features of Notes and Domino, moving responsibility for such features from the application layer (Notes) to the security-device layer (the smartcard itself). In particular:

- Enable *Notes ID recovery* (discussed extensively in 3.3, “How ID recovery works” on page 35) on an ID file *before* smartcard-securing the file, because this allows the ID to be recovered back to passphrase-only state if the smartcard is lost or destroyed, including any private keys shifted onto the smartcard. If an organization registers IDs with a certifier already invested with ID recovery information, no steps should be necessary prior to smartcard enablement.

  Adding a new or resetting an ID recovery payload (“information”) *after* smartcard-enablement, while not ideal, can still work,[5] except that private keys pushed prior to smartcard-enablement will not be recoverable directly, but rather only by presentation of the smartcard possessing those keys. In other words, if the smartcard was also lost or destroyed, such private keys are also lost. The only alternative is if the smartcard can be regenerated in the manner described in 4.3.1, “Smartcard-securing a Notes ID” on page 51.

  For the scenario of “resetting” the recovery payload (as opposed to the scenario of creating the payload a first time), we can further qualify that because ID recovery in

[5] Contradictory documentation elsewhere aside (including on the User Security dialog box itself).

practice sends an encrypted copy of an ID to the organization's ID repository whenever a protected component changes (for example, a special symmetric encryption key added), smartcard-resident private keys can be located in copies of the ID preserved in the repository, and so able to be painstakingly restored to a recovered ID.

- Disable Notes single logon by clearing the **Login to Notes using your operating system login** option on the Security Basics panel of the User Security dialog box. Any possible single-logon function will be managed by the smartcard device and its software drivers. PKCS #11 intentionally provides no cross-application single sign-on (SSO) support. Notes attempts nothing proprietary on this front.
- Server-based *password expiration* must be disabled on the Person documents of smartcard-enabled users; otherwise, users will become locked out from accessing their Domino servers. To disable password expiration, on the user's Person document in the Domino Directory, set the Required change interval field to zero (in the People view → edit Person document → Administration tab → Password Management section).
- Server-based *password checking* is supported with smartcards if all ID files extant are file system copies of the ID secured by the smartcard (contrary documentation in Notes online help aside). Similar to the case of password expiration, password checking is controlled on a user's Person document in the Domino Directory by activating **Check password** in the Password Management section.
- Next, smartcard support does not yet extend to the registration of new users and servers. Therefore, security architects and administrators should include an enablement step in deployment plans: manual one-by-one or automated by a Notes C API program.

5

Enhancements for longer keys in certificates and IDs

In this chapter, we discuss the enhancements made to the Notes public key infrastructure (PKI) in Lotus Notes and Domino 7. You can use private keys to make authentication and encryption, the foundations of Notes and Domino security, even more secure than they already are. We discuss the following topics:

- How Notes and Domino use PKI
- How longer keys enhance security
- ID and key maintenance issues
- Key rollover in Notes and Domino 7 for clients and servers

5.1 How Notes and Domino use public key infrastructure

Lotus Notes was designed to enable users to collaborate and share information. One of the key features that helped Notes succeed at this was security. In order to get people to trust a system that was designed for sharing information, they had to trust that Notes could and would limit sharing so that information would not be accessed by people who were not supposed to see or change it.

By convention, computer security professionals like to divide the world into “good guys” who want to protect their systems and information, and “bad guys” who want to break in and read confidential data. One of the ways good guys protect their systems is with cryptography, which uses mathematical techniques to transform readable data, known as plaintext, into unreadable data, known as cryptotext. There are two main branches of cryptography, “public key” and “secret key,” also known as “asymmetric” and “symmetric,” respectively. Lotus Notes and Domino use both public and secret key techniques, and like most other practical security systems, it combines them into a hybrid, because a pure public key system for messaging is inefficient.

Note: Public key encryption uses pairs of keys, known as a “public/private key pair,” but for most purposes of this discussion, we can dispense with that formality and refer only to public keys. When we talk about the strength of a public key, we really are talking about the strength of the public/private key pair.

Another term, “bulk data key,” refers to a secret key that is used one time to encrypt a document or message. For each recipient of the message, a copy of the same bulk data key is encrypted using public key encyrption. The security of each individual message sent in this manner is determined by the strength of the bulk data key. The security of the system is determined by the strength of the public keys.

The cornerstone of a trustworthy computer security system is authentication, the process of determining the identity of users, and also servers, so that restrictions on their access can be enforced. There are many different ways of doing authentication, some of which are comparatively weak, such as prompting for a name and password, and some of which are much stronger. Lotus chose to implement stronger authentication using public key cryptography. Certificate and ID files and the Name and Address Book (now known as the Domino Directory) in Notes have provided secure authentication for Notes and Domino since the very first release. Together, these components make up the Notes and Domino public key infrastructure, or PKI for short.

A brief explanation of how Notes and Domino use PKI

The mathematics behind public key cryptography is beyond the scope of this book, but we list the following basic principles behind the Domino PKI:

- There are three types of entities for which PKI authenticates and helps protect data:
  - Certifiers
  - Servers
  - Users
- Each entity has two mathematically related keys, which are really just very big numbers:
  - The public key
  - The private key

- Although the two keys are related mathematically, it would take a very long amount of time and a lot of computer power to figure out either one from the other.

  The more digits (or bits) in the keys' really big numbers, the longer the time and computer power, thus the stronger the authentication and protection are.
- Notes and Domino use the RSA algorithm for all operations with public and private keys.
- An entity's private key is kept securely locked inside its ID file, and it is used for:
  - Signing things that the entity is sending to other others.
  - Decrypting bulk data keys that encrypt messages others have sent to the entity.
- An entity's public key is kept in the Domino Directory, where anyone can access it, and it is used for:
  - Encrypting a bulk data that another entity is using to sending a message to the entity.
  - Validating the signature on things the entity has received from others.
- Authentication is based on the security of digital signatures:
  - Certifiers sign certificates that are added to server and user IDs.
  - Servers and users exchange signed certificates to establish a trust relationship, and they then sign credentials to prove that they are who they claim they are.
- Encryption is a four-step operation:

  a. Creating a "bulk data" key, which is a much smaller number than the public and private keys.

  b. Encrypting data using the short key and a cipher algorithm:
     - RC2 is the block cipher used for field encryption.
     - RC4 is the stream cipher used for network port encryption.

  c. Encrypting the short key using the public key of the recipient as the RSA public key encryption algorithm.

     This step is repeated for each recipient of the encrypted data.

  d. Combining the encrypted data with the encrypted key so that it can be sent to all recipients.

5.2 How 1024-bit keys enhance security in Notes and Domino 7

In both secret key and public key cryptography, the more bits (or digits) that the good guys use in public and private keys, the harder it is for bad guys to break in. Secret keys require many fewer bits for the same level of security as public keys. These days, 128 bits or more are the norm for secret keys, and 1024 bits or more are recommended for public keys.

The slow speed of computers in the late 1980s also put practical limitations on the size of keys that could be used in early versions of Notes and Domino without significant performance penalties. In addition, until January 2000, Notes and Domino were required to conform to restrictions imposed by the United States government, which limited the size of public and private keys and bulk data keys in versions that were exported outside of the U.S., so there were North American and International versions of Notes and Domino that worked with different key sizes. There was even a special version of Notes and Domino just for France, because French import restrictions did not allow use of the "workfactor reduction" scheme that IBM negotiated with the U.S. government to allow the export of longer keys to other countries.

Despite these practical and legal limitations, the key sizes the good guys have been using in Notes and Domino until now are, for almost all business purposes, still strong enough to withstand even expensive and sophisticated attacks by the bad guys. Unfortunately, that will not remain true for much longer, and for some customers, it is already not true.

One way of thinking about this is that the Notes PKI had roughly 20 to 25 years worth of solid protection designed into it, but Notes is 20 years old now. From a purely technical perspective, the period of protection has already run out for good guys who started with the earlier International version of Notes and Domino, and by the end of this decade, it is likely that the protection for all Notes and Domino users will be in danger of running out.

From a business perspective, the current situation is not quite so dire, because bad guys still face significant costs in order to break into data protected even by the weaker International keys used in the Notes PKI. Security professionals, however, do not like making close judgements about the value of good guys' data and how much bad guys might be willing to spend in order to steal it. They want the costs to the bad guys to always be far too outrageous to even be considered, and by that standard, Notes and Domino need an upgrade.

Important: Even as this book was being written, new advances in cryptography have been announced. A team of researchers in Germany has published results of a successful attack on a 640-bit RSA public key. The attack took three months and required the continuous use of 80 processors. While this is still an expensive and difficult task, it proves that attacks on the public keys in Notes 6.5 and earlier are technically feasible today.

In Notes and Domino 7, IBM has taken steps to make Notes and Domino meet the standards of security professionals by migrating from the current key sizes to larger keys. As of Notes and Domino 7, keys of 1024 bits are fully supported, and the support for even larger 2048-bit keys is ready and waiting to be turned on by a future release.

5.2.1 Examining your ID files to find out what strength your keys are now

In Domino Administrator, go to the Configuration tab and click **ID Properties** in the Certification section. Select an ID file, your certifier ID, for example, and enter its password. In the ID Properties dialog box, click **Your Identity** and then **Your Certificates**. You should see one or more hierarchical certificate entries in the table within the panel, as shown in Figure 5-1 on page 65.

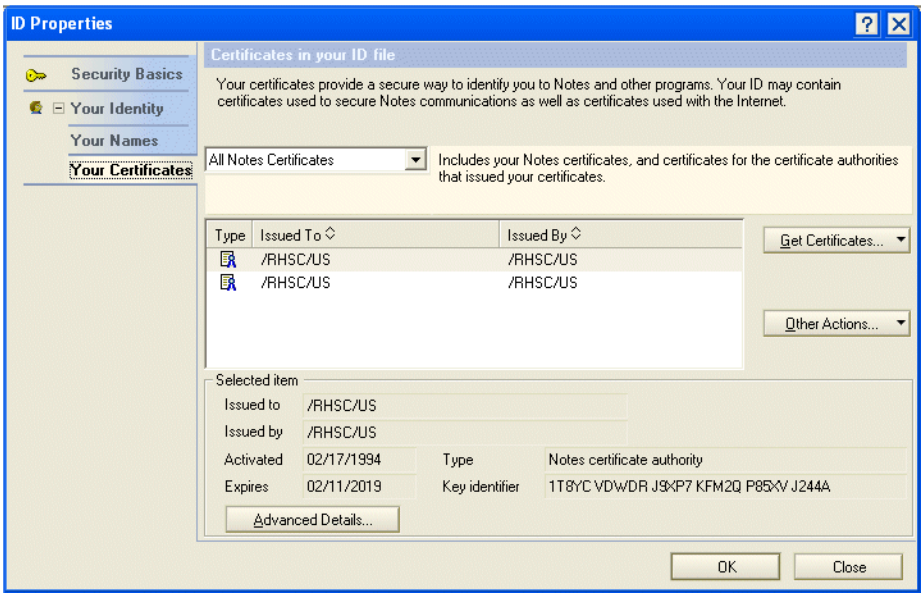


Figure 5-1 Examining ID file properties

Select each of the hierarchical certificates in turn, and click **Advanced Details**. The last line of information in the Notes Certificate Advanced Details dialog box shows a key strength of either 630 or 512 bits, as shown in Figure 5-2. You might have both, in which case, the longer key will be used whenever possible.

Note: In some organizations, a mixture of North American and International IDs might have been created. In fact, in some organizations, the practice of using North American certifier IDs but creating all server and user IDs with International keys was adopted, so it is worth spot-checking a few user IDs and server IDs as well as your certifier ID.

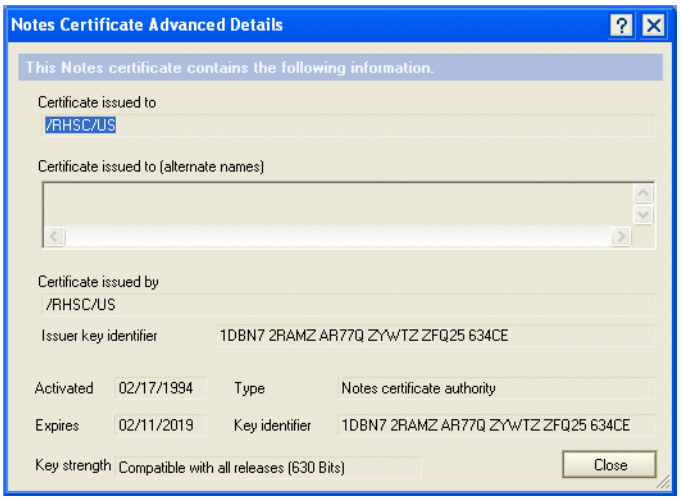


Figure 5-2 Notes Certificate Advanced Details

Note: On the Security Basics tab of the ID Properties dialog box, the ID file encryption strength is most likely shown as 64 bits. This indicates that the ID file itself is encrypted with a 64-bit "password-derived key," which is created from a hash of your password. This is unrelated to the key lengths used by Notes and Domino for other purposes, but it is part of the overall security picture. The ID files contain private keys, so the level of protection given by the password and encryption algorithm are an important factor in keeping the private keys secure. In Notes and Domino 7, 128-bit keys for protection of ID files are available, but earlier versions of Notes can only read ID files that have 64-bit password-derived keys. When registering ID files in Domino Administrator 7, you can choose the strength of the password-derived key that is used by selecting from the drop-down list in the Password Options dialog box, as shown in Figure 5-3.

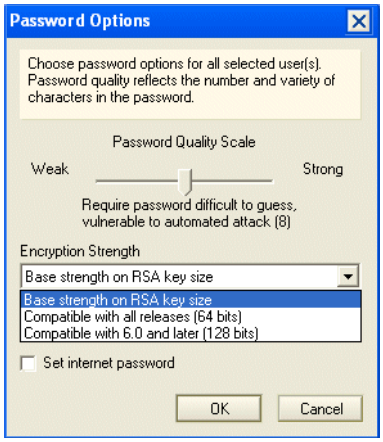


Figure 5-3 Choosing password strength

5.2.2 Forward and backward compatibility

Historically, two factors limited the key sizes that any version of Notes and Domino would use. One factor is the encryption software that is included in the core API code that is included in the particular release. The other factor was the restrictions that the U.S. government placed on what software exported from the United States could do. Fortunately, the way these legal restrictions worked, interoperability between software running in the U.S. and software running outside the U.S. was allowed. For this to be possible, the Notes and Domino code has drawn a distinction between the key size used for data that it generates and the key size for data that it accepts. The International versions of Notes and Domino do not generate data using key sizes larger than allowed by U.S. export laws, but can accept data using key sizes up to the maximum supported by the encryption API included in the release. Export restrictions are no longer a factor, but this same strategy of supporting larger key sizes when accepting data has been used in Domino 6 and 6.5 to prepare for the support of larger keys in Domino 7, and it is also being used in Domino 7 to prepare for even larger key sizes in future releases.

5.2.3 Key sizes in early Notes and Domino versions

Versions of Notes and Domino prior to Release 6 created and used keys with the sizes shown in Table 5-1 on page 67.

Table 5-1 Key sizes in early Notes and Domino versions

Key	Size
Keys generated by Notes 3	
Public and private key size	512 bits Generates 64 bits (North American) or 40 bits (International)
Bulk data key size	64 (North American) 40 bits (International)
Keys accepted by Notes 3	
Public and private key size	Up to 760 bit public keys, and up to 630 bit private keys
Bulk data key size	64 (North American) 40 bits (International)
Keys generated by Notes and Domino 4	
Public and private key size	630 bits (North American) or 512 bits (International and French)
Bulk data key size	64 bits (North American and International) or 40 bits (French) Note that in the International version the 64-bit keys offer only 40 bits of protection against U.S. government agencies.
Keys accepted by Notes and Domino 4	
Public and private key size	Up to 760 bits
Bulk data key size	64 or 40 bits
Keys generated by Notes and Domino 5	
Public and private key size	630 bits (North American) and 512 bits (International)
Bulk data key size	64 bits
Keys accepted by Notes and Domino 5	
Public and private key size	760 bits
Bulk data key size	64 or 40 bits

5.2.4 Key sizes in Notes and Domino 6.x and 6.5x

Due to the relaxation of U.S. export laws, there was no division between the North American and International versions of Notes and Domino in Releases 6 and 6.5 of Domino. That simplifies matters considerably, but customers who built their infrastructures initially with international keys had no easy upgrade path to move to stronger keys.

In anticipation of the need to upgrade the Notes and Domino PKI to larger and thus more secure keys in Release 7, forward-compatibility support was built into Notes and Domino 6.

Release 6 can recognize and use larger keys, but it cannot create them. See Table 5-2 on page 68.

Table 5-2 Key sizes in Notes and Domino 6.x and 6.5x

Key	Size
Keys created by Notes and Domino 6.x	
Public and private key size	630 bits (North American) or 512 bits (International)
Bulk data key size	64 bits
Keys that can be used by Notes and Domino 6.x	
Public and private key size	1024 bits, 630 bits, or 512 bits
Bulk data key size	128 bits (6.04 or later), 64 bits, or 40 bits

5.2.5 Key sizes in Notes and Domino 7

Release 7 is the first version of Notes and Domino that can create long keys. It understands and can use the shorter keys created by all previous versions, and it also understands and can use even longer keys that will be supported by a future release. See Table 5-3.

Table 5-3 Key sizes in Notes and Domino 7

Key	Size
Keys created By Notes and Domino 7	
Public and private key size	1024 bits or 630 bits
Bulk data key size	128 bits
Keys that can be used by Notes and Domino 7	
Public and private key size	2048 bits, 1024 bits, 630 bits, or 512 bits
Bulk data key size	128 bits, 64 bits, or 40 bits

5.2.6 Environments with mixed software versions and key lengths

ID files created with Notes and Domino 7 that have 1024-bit keys are backward compatible with Notes and Domino 6.x, but they *cannot* be used by Notes 5 clients or a Domino 5 server.

A network of clients and servers with a mixture of Release 5, 6, and 7 clients, and servers, however, can interoperate as long as the R5 clients and servers use ID files that were created with software from Release 6.x or earlier. An ID created by Notes and Domino 7 with a 1024-bit key will also contain a 630-bit key, and this makes network compatibility possible. All releases of Notes and Domino recognize the lengths of the public keys of other users and servers that they connect to or exchange data with, and will use its own key of with the largest matching length.

5.3 ID and key maintenance: Creating new IDs with long keys in Notes and Domino 7

When registering certifiers or users in Notes and Domino 7, you need to choose the length for the public key. The default will be 630 bits, which is compatible with all earlier releases. You can change this to 1024 bits if your environment consists entirely of Notes 6, 6.5, and 7 clients and the same releases for your Domino servers. Figure 5-4 shows the choice between 630-bit and 1024-bit keys in the Public key specification field in the Register Organization Certifier dialog box, and Figure 5-5 on page 70 shows the choice between 1024-bit keys and 630-bit keys in the Public key specification field on the ID Info tab in the Register Person dialog box. (The ID Info tab is only available after you select the **Advanced registration features** option.)

Note: Customers who are new to Notes and Domino should use 1024-bit keys for their certifier IDs, but customers with existing Notes and Domino installations can still use an existing certifier to create ID files with a 1024-bit key even though the certifier has a 630 or 512-bit key. Smaller organizations with very high security needs might want to consider recertifying all IDs using a 1024-bit certifier, but for most organizations, the costs of this will outweigh the benefits. Shorter keys on certifier IDs are a theoretical weakness that could allow bad guys to create their own ID files that appear to be legitimate, but the barriers to this are still very high. Furthermore, the Notes and Domino password and key checking features can be used to provide protection against this type of attack.

Physical security and a strong password on the certifier ID file itself, whatever the key length, remain the most important factors in the security of your Notes and Domino environment.

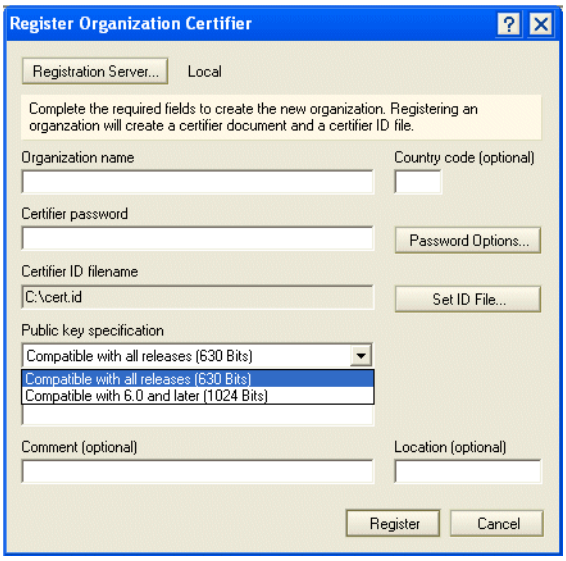


Figure 5-4 Register Organization Certifier with a 1024-bit public key

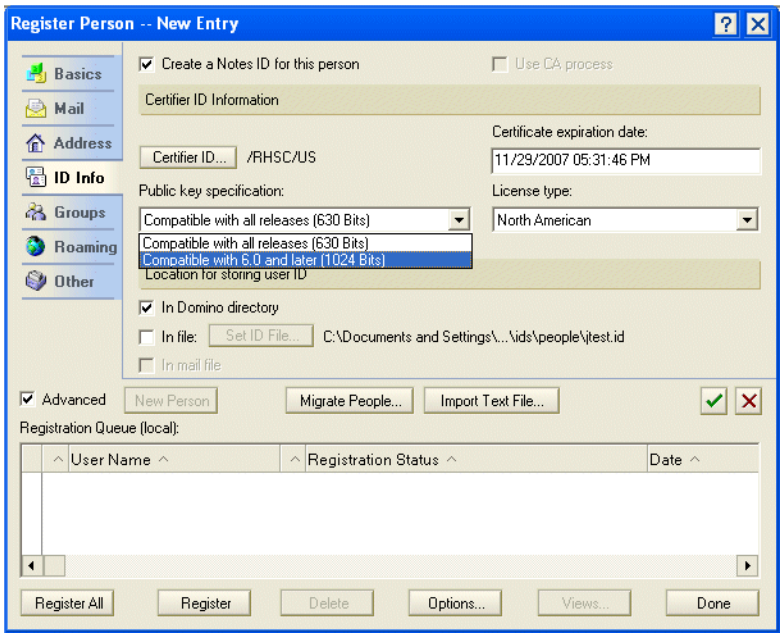


Figure 5-5 Register Person with a 1024-bit public key

5.4 User and server key rollover

Key rollover is the process used to update the set of Notes public and private keys that is stored in user and server ID files. Key rollover can be used to upgrade to 1024-bit keys from shorter keys. It can also be used to change public keys in the event of a security breach that compromises a user's ID file. Key rollover for users can be requested by individual users, or it can be initiated on behalf of groups of users by an administrator using policies. Key rollover for servers is initiated by an administrator using fields in the Server document.

Important: Unlike processes available for recertifying an ID with a new public key in previous versions of Notes and Domino, the key rollover process in Notes and Domino 7 preserves an archive of the previous public/private key pair in the user's ID file. This enables users to continue to read mail and documents that were encrypted using their old public key.

5.4.1 Manual key rollover

A user can initiate key rollover by opening the User Security dialog box and selecting **Other Actions → Create New Public keys** in the Your Certificates tab. This opens the Create New Public Keys dialog box, as shown in Figure 5-6 on page 71. Leave all settings at the default values, and click **Create Keys**. (Changing the Request Certificate Using drop-down list causes the request to be handled through the mail-based process available in earlier releases instead of using the new rollover process.) The next time the user connects to the home server, a prompt opens asking whether they want to create a copy of their ID file. Users who work on multiple PCs should make the copy and use it to replace their ID file on the other machines.

Important: Users should *not* request new public keys from each of their computers. Administrators should monitor ADMIN4.NSF for repeated key requests from users. Multiple requests indicate a high probability of a confused user who is about to get into a situation where ID files on different computers have different keys.

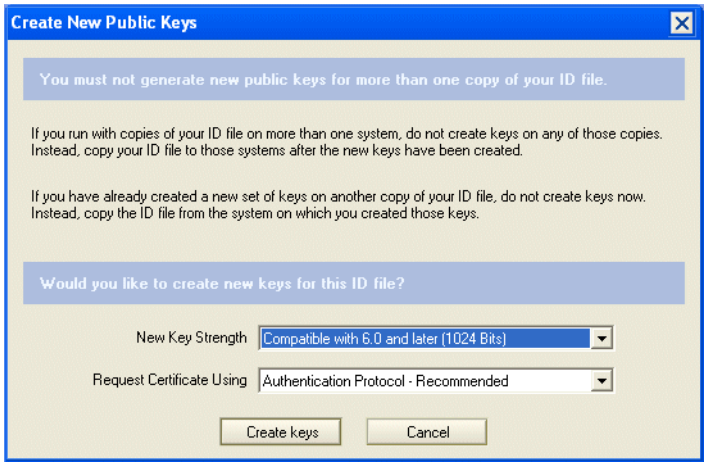


Figure 5-6 User-initiated key rollover

When the rollover process is initiated, a new public/private key pair is added to the user's ID file, but it does not replace the existing key. It is marked as "pending," and a request to certify the new key is posted to the ADMIN4.NSF database on the server. An administrator must certify the new key before it can be used. Figure 5-7 on page 72 shows the certify new key requests view in ADMIN4. The administrator uses the Certify Selected Entries button in this view to process the rollover requests. This generates a Recertify Person In Domino Directory request in ADMIN4, which the adminp process on the domain's administration server will take care of when it processes its queue.

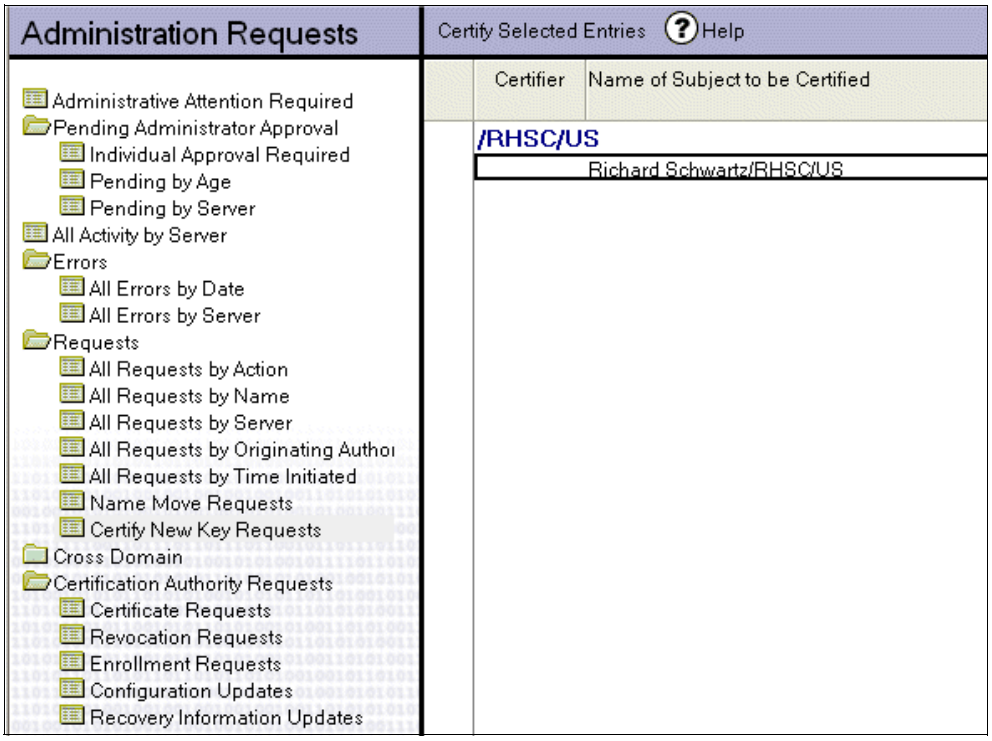


Figure 5-7 Certify new key requests

Note: If the user's home server is not the administration server for the Domino domain, the adminp request will have to replicate and the new key in the Person document will have to replicate back before the new key will be available.

When adminp finishes with this, the new key information will be in the user's Person document, and the next time the user authenticates with the home server, a dialog box opens asking if the user want to accept the new key to complete the rollover process, as shown in Figure 5-8 on page 73. Users who have more than one PC will see this dialog box on each PC.

Tip: If a user with more than one PC does not see the Accept New ID Information dialog box on each PC, this is an indication that the copy of the ID file that should have been made at the beginning of the rollover process was not properly copied to other PCs.

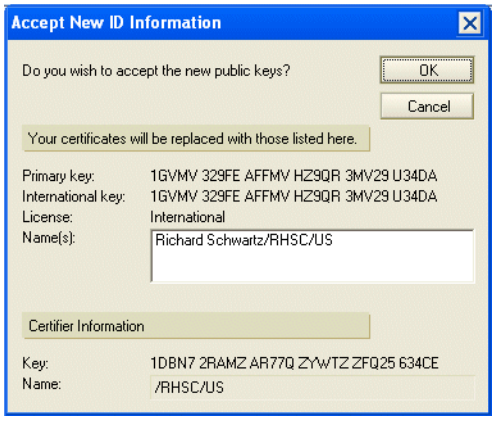


Figure 5-8 Completing the key rollover

5.4.2 Policy-based key rollover

Very few users can be counted on to be aware of the significance of upgrading to 1024-bit keys, so relying on the manual rollover process alone to improve the security of your Notes and Domino infrastructure is problematic. Fortunately, administrators can use policies to trigger rollover for users. To do this, start by creating a security settings document using the Domino 7 Administrator client, as shown in Figure 5-9 on page 74. If you are already using policies, you might prefer to modify existing security settings documents instead of creating a new one.

Note: For a thorough overview on Policies, refer to 2.1, “Policies and policy settings” on page 8.

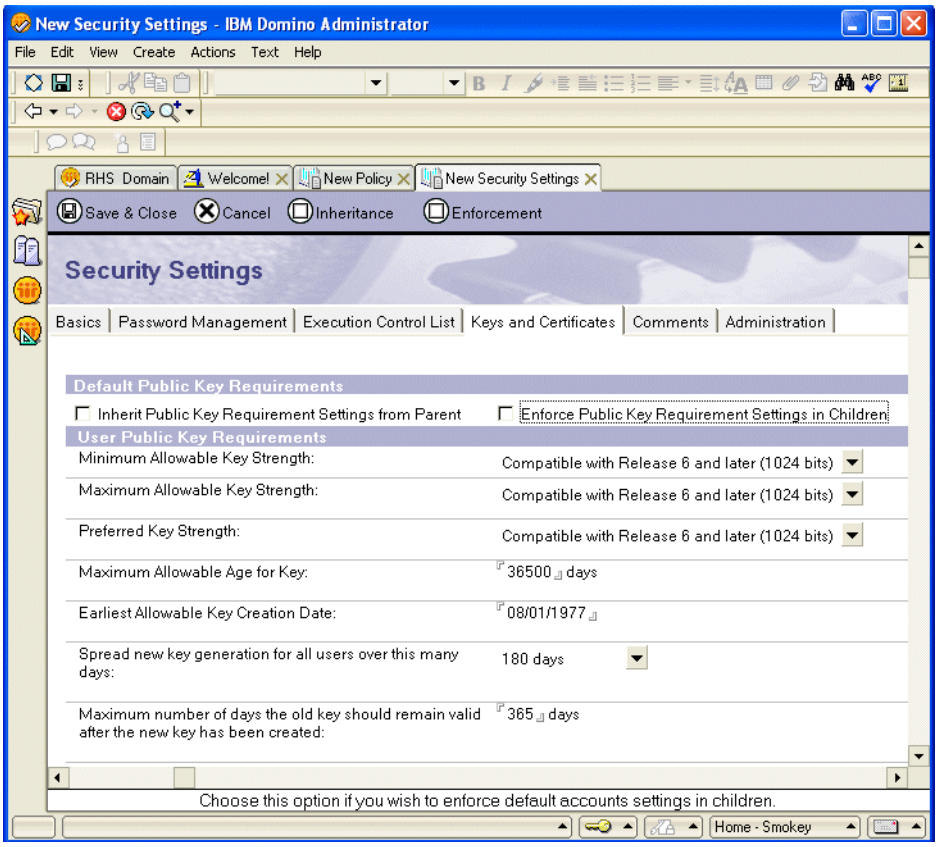


Figure 5-9 Security Settings

On the Keys And Certificates tab of the security settings document, you can set the minimum, maximum, and preferred key strength. When you want to do a rollover, set all of these to 1024 bits.

Important: In practice, setting the minimum key length to 1024 is not always sufficient to force an immediate rollover all by itself. To be sure that a rollover will occur, set the maximum allowable age for the key or the earliest allowable creation date for the key in order to force the issue. The creation date is easier to work with. If you know the date that your most recent users were certified, you can set the earliest allowable date to be one day after that date.

The security settings document must be attached to a policy in order to have any effect. If you are using policies already, open an existing explicit or organizational policy and add the name of the security settings document that you just created to the Security field.

Note: If you are using explicit policies, you will have to add the policy's name to the Assigned Policy field on the Administration tab of the Person document for each user that you want to roll over.

After you create and put into effect the policy settings for the rollover, users will be prompted to initiate the rollover when they authenticate with their home server. From this point, the process proceeds in the same manner as the manual process.

5.4.3 Server key rollover

Server ID files and the keys they contain are second only to certifier ID files in their importance for the overall security of your Notes and Domino infrastructure. If a server ID is compromised, a great deal of data will be at risk. The good news is that upgrading server IDs to 1024-bit keys after you upgrade to Domino 7 is essentially painless.

Important: Even with an upgrade to 1024-bit keys, server IDs in many Domino environments have minimal protection. They are often not even protected by a password. Since Release 6, it has been possible to configure Domino servers to automatically restart after a crash, and this works without requiring re-entry of the server ID password. If you are concerned about the security of your server IDs, set passwords on them and enable fault recovery in the Server documents in addition to upgrading them to 1024-bit keys.

Because policies apply only to users, not servers, and because servers cannot initiate the manual rollover process, administrators must do it for them. This is done through a set of fields on the Administration tab of the Server document, as shown in Figure 5-10. The fields are similar to the ones in the security settings document for policy-based rollovers.

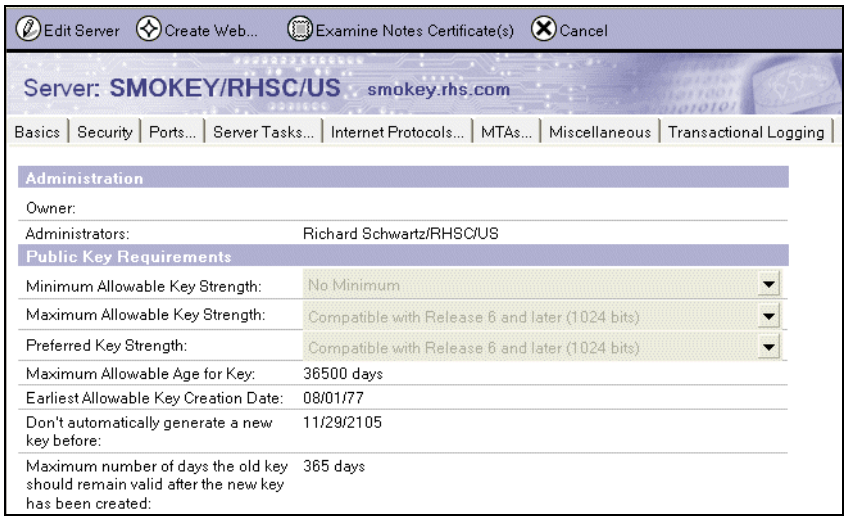


Figure 5-10 Rollover settings in the Server document

Tip: If you are only rolling over one server, make the changes to the Server document in the replica of the Domino Directory that is on that server. If you are rolling over several servers, make the changes to the Domino Directory on one server and make sure that the changes replicate to all the other servers before continuing with the rollover process.

The rest of the server key rollover process is similar to the rollover for users, with the primary difference being that the actions that take place when a user authenticates will instead take place during a server restart. It takes two restarts of the server to complete the process. The first restart causes the new key to be generated and saved in the server ID file, marked as “pending.” A certification request is simultaneously written to ADMIN4.NSF. As with a user rollover, the certification request must be processed by an administrator, and then a second request is processed by adminp on the domain’s administration server in order to add the new keys to the Server document.

Tip: You can use the following command sequence on the server console of the domain's administration server in order to expedite handling of the adminp request:

```
rep <server-name> names.nsf
rep <server-name> admin4.nsf
tell adminp process all
rep <server-name> admin4.nsf
rep <server-name> names.nsf
```

The order of commands is important. Replicating ADMIN4.NSF before NAMES.NSF can result in a replication conflict in the Server document if adminp actually processes the second request before NAMES.NSF replicates from the server where you entered the rollover settings.

The **rep** commands are not necessary if the server whose ID you are rolling over is the administration server for your domain.

After the replication of the Domino Directory brings the updated Server document form the administration server to the server that is being rolled over, the server can be restarted again. During startup, the server queries the Server document and finds the certificate for its new key. The new 1024-bit key is activated and the rollover is complete.

5.5 Public key checking in Notes and Domino 7: Validation and authentication

Whenever a Notes client or Domino server attempts to communicate with a Domino server to replicate, route mail, or to access a database, two security procedures, validation and authentication, use information from the client or server ID to verify that the client or server is legitimate. Validation establishes trust of the client's public key. If validation occurs successfully, authentication begins. Authentication verifies the user's identity and uses the public and private keys of both the client and the server in a challenge/response interaction.

Rules that guide trust of public keys

Validation uses these three rules to establish the trust of a public key. Domino validates the client that is trying to access the server and the server that the client is trying to access.

- Trust the public key of any of the server or client's ancestors in the hierarchical name tree because the ancestor's public key is stored in the server or client's ID file.
- Trust any public key obtained from a valid certificate issued by any of the server or client's ancestors in the hierarchical name tree.
- Trust any public key certified by any trusted certifier and belonging to one of the certifier's descendants.

How validation and authentication work

This example describes how validation and authentication work together to ensure the security of the system. In this example, user Randi Bowker/Marketing/East/Acme (the client) wants to access Mail-E/East/Acme (the server).

1. Mail-E reads the Acme public key from the Mail-E ID file. According to the first rule just described, Mail-E trusts the public key assigned to Acme.

2. Randi sends Mail-E information in her user ID. Mail-E reads Randi's user ID for the certificate issued by Acme to East. Mail-E uses the Acme public key, which it now trusts, to verify that the East certificate is valid. According to the second rule, if the certificate is valid, Mail-E trusts the public key assigned to East.
3. Mail-E then reads Randi's user ID for the certificate issued by East/Acme to Marketing. Mail-E uses the East/Acme public key to verify that the Marketing/East/Acme certificate is valid. Again, the second rule states that Mail-E now trusts the public key assigned to Marketing/East/Acme.
4. Mail-E reads Randi's user ID for the certificate issued by Marketing/East/Acme to Randi. Mail-E uses the Marketing/East/Acme public key, which it now trusts, to verify that Randi's certificate is valid. According to the third rule, if the certificate is valid, Mail-E trusts the public key assigned to Randi.
5. After Mail-E establishes trust of Randi's public key, the authentication process begins.
6. Mail-E sends a random number challenge to Randi.
7. Randi's workstation encrypts the challenge with her private key and sends the newly encrypted number back to Mail-E.
8. Mail-E uses Randi's public key to decrypt the response. If this yields the original challenge, Mail-E knows Randi is who she claims to be.
9. The process is then reversed. Randi's workstation validates Mail-E's public key by processing Mail-E's certificates and then uses the challenge/response procedure just described to authenticate the server.

Public key checking

The signatures on user and server certificates exchanged during authentication are always checked. You can enable an additional level of verification for public keys by having the value of the public key passed in the certificates checked against the value of the public key listed in the Domino Directory. It is possible for users to authenticate with a server, but to have a mismatch between the value of the public keys in their certificates and the value of the key that is listed for them in the Domino Directory.

This extra level of key verification protects against misuse of a lost or compromised ID file. Typically, if an ID file is lost, its owner needs to be registered to create a new ID file and directory entry. If the ID file has been compromised, the owner's public and private keys need to be rolled over (see 5.4, “User and server key rollover” on page 70 for more information about key rollover) and that new set of keys need to be certified (thus updating the directory entry). By enabling directory-level key checking, an attacker in possession of the old ID file will not be able to use it to access the server, even though that old ID file might contain a valid certificate.

A new enhancement for Domino 7 is the option to enforce key checking for Notes users and Domino servers listed in trusted directories only**.** This option allows administrators to give users not listed in the directory access to databases and applications on the server. For example, a database might have its access control list configured to give Editor access enabled for users listed in the Domino Directory, and Reader access for everyone else. So if this key checking option is enabled, users not listed in the directory can still access the server to use the database for which they will have Reader access only.

As of Domino 7, you can also choose to control whether a log message is generated if authentication succeeds but a mismatch is detected. This enables administrators to detect when the ID file contents have gotten out of sync with directory entries, but to do so without preventing those users from authenticating because of public key mismatches.

Public key checking is enabled in the Server document on the Security tab. Click the drop-down list next to “Compare public keys” and choose one of the following options:

- Enforce key checking for all Notes users and Domino servers: Compares the key value in the certificates passed during authentication against the key value stored in the Domino Directory. Any user or server not listed in a trusted directory will be treated as though it failed this verification check and will not be allowed to access this server.
- Enforce key checking for Notes users and Domino servers listed in trusted directories only: Compares the key value in the certificates passed during authentication against the key value stored in the directory only when the user or server is listed in a trusted directory. Any user or server not listed in a trusted directory will be treated as though it passed this verification check.
- Do not enforce key checking: If you want only the certificate signatures checked during authentication, but not verify the keys against the directory contents.

Click the drop-down list next to “Log public key mismatches” and choose one of the following options:

- Log key mismatches for all Notes users and Domino servers: Logs events that occur when the key value in the certificates passed during authentication does not match the key value stored in the Domino Directory.
- Log key mismatches for Notes users and Domino servers listed in trusted directories only: Logs events that occur when the key value in the certificate passed during authentication does not match the key value stored in the directory only when the user or server is listed in a trusted directory.
- Do not log key mismatches: Logs only authentication failures.

Note: You must restart the server so that the changes take effect. The server polls every hour to see if these settings have changed, so if the server is not restarted, it might be as long as an hour before the new settings take effect.

6

Single sign-on (SSO) and name mapping in Domino

This chapter provides the how-to information for configuring single sign-on (SSO) with the new user name mapping feature. SSO is not new to Domino. Release 7 introduces the ability to chose the name that is stored in the Lightweight Third Party Authentication (LTPA) token. LTPA was designed by IBM and is a de facto standard across the IBM product family.

As long as we authenticate with only one server, we do not have problems passing credentials (user ID and password) from one system to another. When we access other systems without being prompted for an ID and password, SSO becomes important. Different systems might not share the same directory and the identification or user name can be different. A Domino user name might not map to a WebSphere user name. With the name mapping option in Domino 7, administrators have more control over the name that is passed between systems.

The vehicle to carry the user information between systems is the browser cookie, also know by LTPA token. It carries a digital signature, creation date, expiry date, and the user name. A cookie by definition is only known within one domain. Therefore, a cookie created for ibm.com® will not be available to applications from emailreplies.com.

This chapter reviews SSO and the new name mapping feature in Domino 7. SSO can be enabled in three different ways: in the Server document, the Web Configuration document, and since Domino R6, in the Internet Site documents. We focus on the Internet Site configuration and the IBM Lotus software product family:

- Lotus Domino Server Release 7
- Lotus QuickPlace® Release 7
- Lotus Sametime® Release 7

To find information about SSO and other IBM products, see the an excellent document just published in September 2005, *Single Sign-on in a Multi-directory World: "Never say login again"* (http://www.ibm.com/developerworks/lotus/library/sso1/). This two-part document explains the processes in great detail. Another source is Chapter 7 in the *Lotus Security Handbook*, SG24-7017, although this does not include R7.

For SSO solutions in a multivendor environment or multidomain environment, you might need to look at third-party tools. One example is from dotNSF (http://dotNSF.com), which uses IBM APIs and allows any combination of these IBM servers (Lotus Domino, Lotus QuickPlace, Lotus Sametime, and IBM WebSphere Application Server and IBM WebSphere Portal, and IBM Tivoli) to natively interoperate with other non-IBM servers, Microsoft Internet Information Services (IIS) among others, in effect making IIS (including integrated win32 authentication) a peer part of the LTPA token cloud of SSO. Generating an LTPA token at the IIS server after Windows integrated silent logon to IIS is a typical example.

This chapter describes an environment with two Domino servers, each in its own domain. One server is the LDAP server and has all the users in the Domino Directory. The second server is an application server with Domino Web applications, Lotus QuickPlace, or Lotus Sametime. Figure 6-1 shows an overview of the test SSO environment.

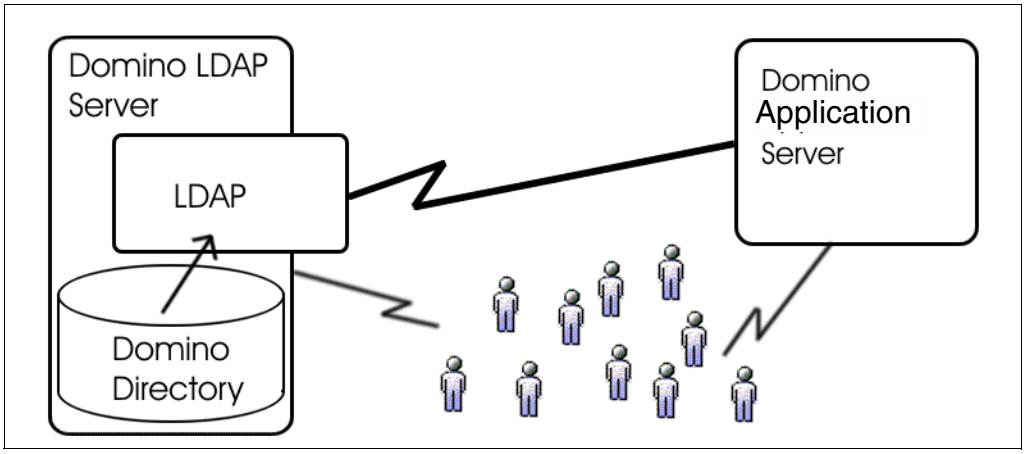


Figure 6-1 Overview of test SSO environment

The purpose is to have two independent Domino domains configured with the new user name mapping feature. This way, users can log in to either of the two servers first. The LTPA token user name is in the foreign name format, not the Notes hierarchical name. The authentication credentials are correctly passed between the servers. All authentication works correctly within the Domino application on both servers.

6.1 User name mapping

The user name mapping feature is disabled by default and requires changes to the Internet Site configuration, the Domino LDAP server, and the Domino directory assistance (or LDAP gateway). Enable the user name mapping feature only if you need to authenticate with a different user name than the Domino hierarchical name.

In Domino, we are known by the cononicalized name (for example, cn=Dieter Stalder/o=Titanium). In other systems, we might be known by another format (for example, uid=ds/o=groofty). In an SSO environment, we have to be known by the name that is accepted by all systems. This shared name is in the LTPA token. Prior to Domino 7, the canonical name was written to the LTPA token when you first authenticated with Domino. In Domino 7, we can choose the name.

Note: Canonical names consist of a list of attributes (for example, cn, ou, o). These attributes are divided by a separator. The Notes separator is a slash, the LDAP separator is a comma.

In our example, we have two Domino servers that are in different domains. One is the LDAP server, the other is the application server. No matter which server we access first, the credentials are correctly created (LTPA token) and passed to the second server.

Important: Names are configured in several places. It is important to use the correct format, Notes or LDAP. Entering a name in the wrong format breaks SSO.

6.1.1 Enable SSO and user name mapping on all servers

These changes apply to all Domino domains that participate in SSO:

- Configuring multiple Domino servers in one domain: This is done once and applies to servers in the domain.
- Configuring multiple Domino servers in multiple domains: This is done once for every domain.

Web SSO configuration

If the Web SSO Configuration document does not yet exist, click **Create Web SSO Configuration**.

Since Release 6, we can configure multiple Internet sites. The different site configurations are kept apart by the organization name. In our examples, we call the organization QP. Figure 6-2 on page 82 shows the Web SSO Configuration document view.

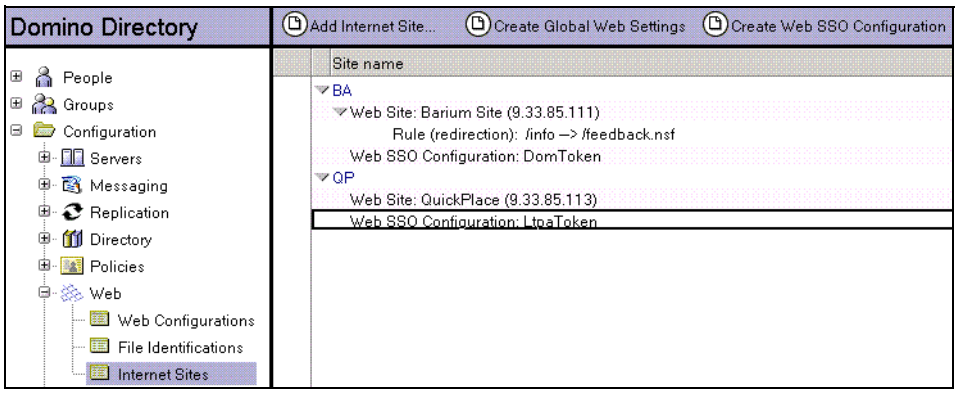


Figure 6-2 View containing Web SSO configuration document

When creating a new configuration, select **Keys** and select **Import WebSphere LTPA Keys**, or in our test setup, we select **Create Domino SSO Key**, as shown in Figure 6-3. This generates the key that will be shared among all systems that participate in SSO. This key is part of the credentials that all the systems share.

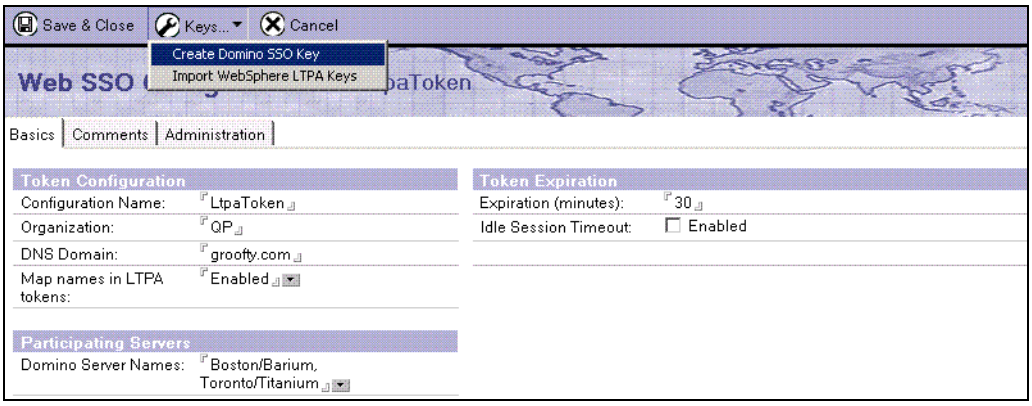


Figure 6-3 Create Domino SSO Key

The window in Figure 6-3 contains the following fields:

- Configuration Name

  This is the name that will be written into the cookie. Use the default name `LtpaToken`. This creates the fewest problems.

- Organization

  Enter a name to represent the configuration. Leaving this field blank will save the document under the view Web Configurations.

- DNS Domain

  Enter the domain for which you set up SSO.

- Map names in LTP tokens

  This field is new in R7. When disabled, the Domino hierarchical name is written in the LTPA token. When enabled, the LTPA user name from the Domino Directory is used (see “Person document” on page 86 for details). To verify the name in the LTPA token, review 6.2.5, “SSO debug instructions” on page 97.

- Domino Server Names

  List all servers that share the same token. The document will be saved and encrypted with the user names and the servers that are listed here. When you also add the foreign servers, you get the warning “Server not found in the Name & Address Book” when saving. Click **OK** to continue. Adding foreign servers to the list is not required, but you need all the servers from the current domain. Keep in mind that you also need to include a user name that will be able to encrypt and decrypt the document in the foreign domain. Cross-certifying the administration ID solves this problem.

Note: When copy and pasting Web SSO Configuration documents from one Domino Domain to another, the document has to be encrypted and saved with the server and administrator public keys. If not, decryption fails on the server and the following error shows in the server log: “HTTP Server: Error loading Web SSO Configuration 'LtpaToken' for Web Site 'QPHome' (Single Sign-On configuration is invalid).

Internet Site: Web

If the Web Site documents do not exist, select **Add Internet Site** → **Web**.

On the Basics tab, provide values for the following fields, as shown in Figure 6-4:

- Descriptive name for this site

  Enter a description for the site.

- Organization

  Enter the same name as from the Web SSO Configuration document.

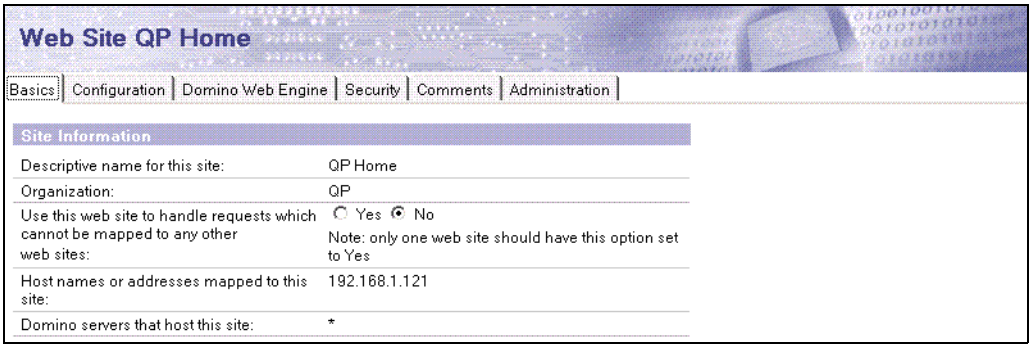


Figure 6-4 Adding a Web Site document

On the Domino Web Engine tab, provide values for the following fields, as shown in Figure 6-5 on page 84:

- Session authentication

  Select the **Multiple Servers (SSO)** option.

- Web SSO Configuration

  Select the token name from the list.

Configure the other options to suit your needs.

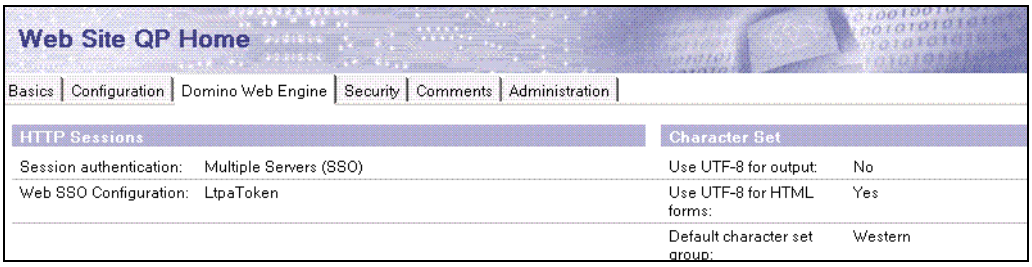


Figure 6-5 Domino Web Engine tab

Activate Internet site configuration

In the Server document, on the Basics tab, enable the field **Load Internet configurations from Server\Internet Sites documents**, as shown in Figure 6-6.

Important: When enabling the Internet Site configuration, some other configurations are disabled. Review the configuration options for Web redirections, IMAP, POP3, IIOP, some SMTP inbound settings, and, most important for our exercise here, the LDAP configuration. The Domino LDAP server needs the LDAP configuration; a missing configuration disables the LDAP server.

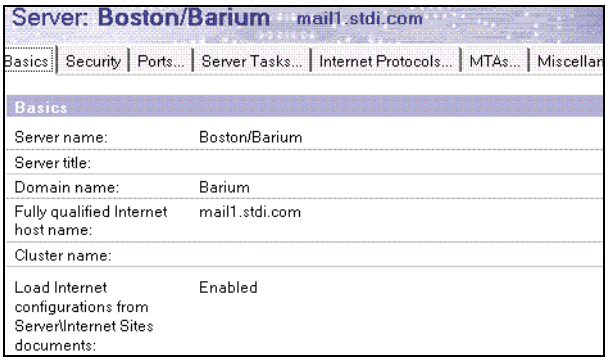


Figure 6-6 Enable Load Internet Configurations from Server\Internet Sites documents

6.1.2 Domino LDAP server configuration

The Domino LDAP server is the system that publishes the Domino Directory. The LDAP server task responds to requests from LDAP clients and the LDAP gateway.

LDAP attribute to Domino fields mapping configuration

To change the default behavior of the LDAP server, we need to modify the default Configuration document. The LDAP server configuration is part of the default Configuration document. The LDAP configuration applies to the whole Domino domain, but the LDAP server task can run on any or all servers. Figure 6-7 on page 85 shows the LDAP server configuration document.

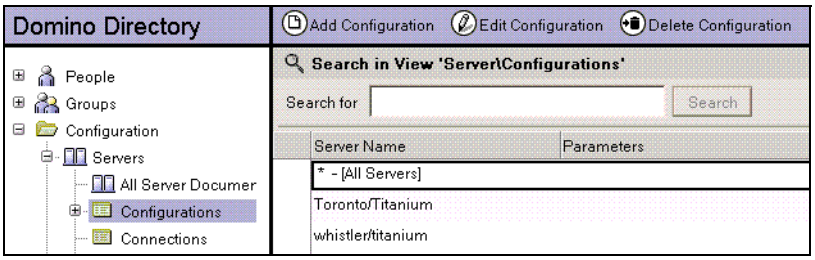


Figure 6-7 LDAP server configuration document

If the default configuration document does not yet exist (* - [All Servers]), click **Add Configuration** to create a new Configuration document and select **Use these settings as the default settings for all servers**, as shown in Figure 6-8.

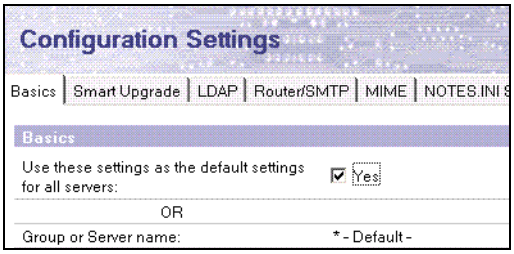


Figure 6-8 Adding a default Configuration document if necessary

Open the LDAP tab in the default Configuration document.

For the name mapping feature to work, one field in the Person document is published in LDAP. From the LDAP point of view, we need to know the Object class, which is called dominoPerson. This is the Person document in the Domino Directory. Then, LDAP needs to know the attribute, which is called LTPA-UsrNm. In Notes terms, this is the field named LTPA_UsrNm in the Person document.

Click the **Select Attribute Types** button. In the LDAP Attribute Type Selection window, select the **dominoPerson** in the Object Classes field and click **Display Attributes**. This displays the list of all possible attributes in the Person document. Scroll down to **LTPA-UsrNm** and select this field, and then click **Add**. Click **OK** to close the window and **Save & Close** the Configuration document. Figure 6-9 on page 86 shows the LTPA-UserNm attribute.

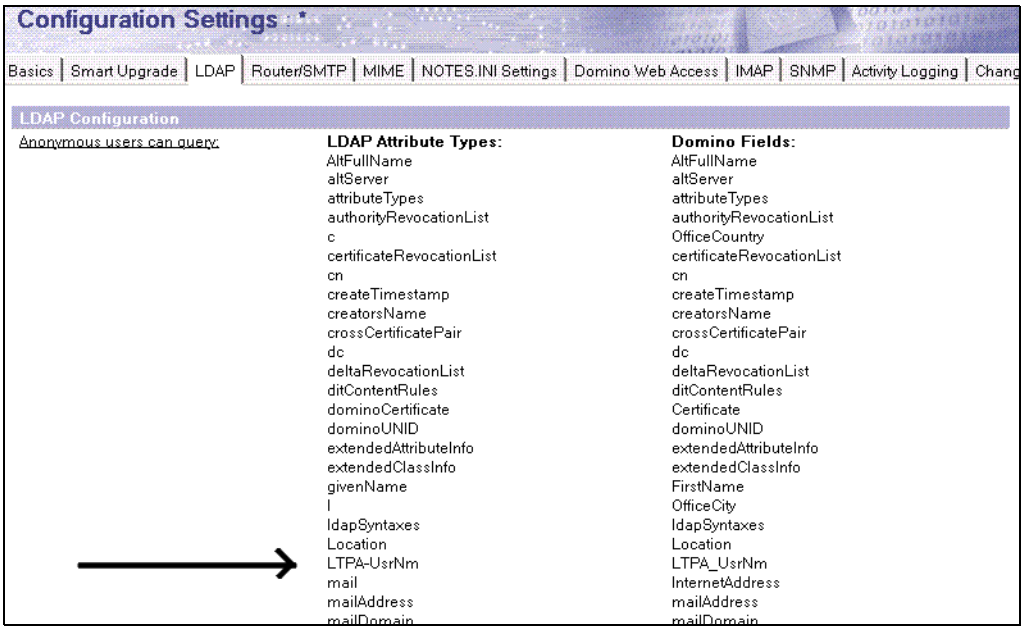


Figure 6-9 List of all possible attributes in the Person document: LTPA-UsrNm attribute

LDAP configuration: De-referencing of alias

The Domino LDAP server only knows the hierarchical Notes name as defined in the first position of the Full Name field in the Person document. To tell Domino to translate from the foreign name (eg uid=ds,o=groofty) to the Notes name (eg cn=Dieter Stalder,o=Titanium), change Allow dereferencing of aliases on search requests? to **Yes**, as shown in Figure 6-10.

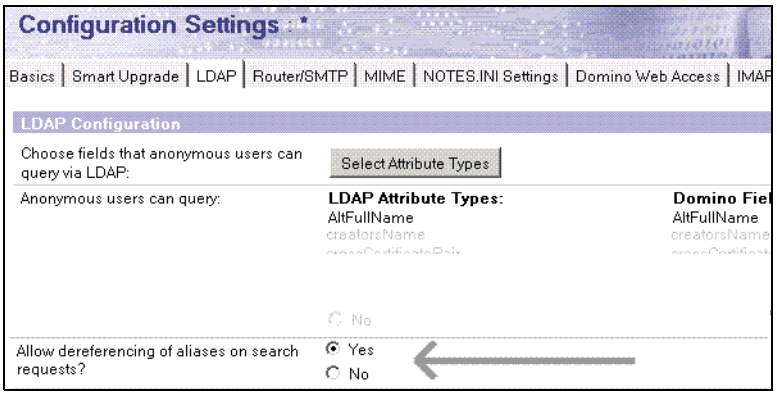


Figure 6-10 Option to de-reference alias

Person document

The Person document in the Domino Directory has a new field, the LTPA user name located on the Administration tab, as shown in Figure 6-11 on page 87.

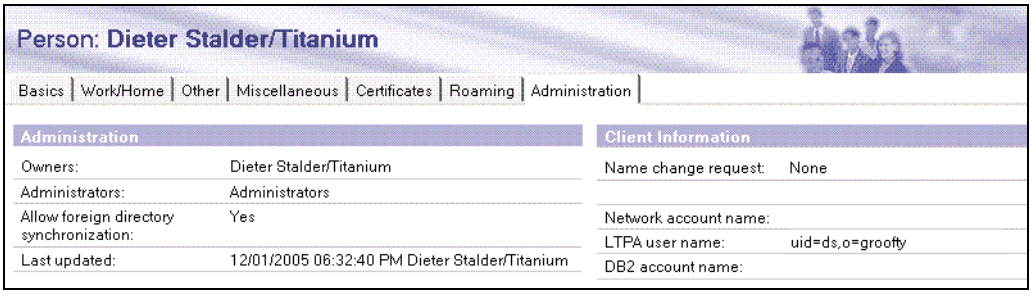


Figure 6-11 In field in Person document in the Domino Directory: LTPA user name

Enter the LTPA user name. The name must be unique to avoid authentication problems. The attributes depend on the systems with which you need to authenticate.

Note: The LTPA user name must be in LDAP format. Attributes are separated by a comma. During the writing of this book, we encountered a discrepancy.

According to the documentation, the LTPA user name should be in Notes format (using a slash). This caused the login to fail in one scenario, when configured with directory assistance (Person document accessed through LDAP). The LTPA token name is CN=dieter instead of UID=ds.

Entering the name in LDAP format (with a comma) worked in all situations. The LTPA token name is UID=ds, the Notes name is correctly translated to cn=dieter, and all read access and ACLs worked correctly.

Enter the name from the LTPA user name field and change to the Notes format (with slash), as shown in Figure 6-12. Also add the unique short name, ds in our example. By adding the name variations to the User name field, Domino will find the users by any of the names.

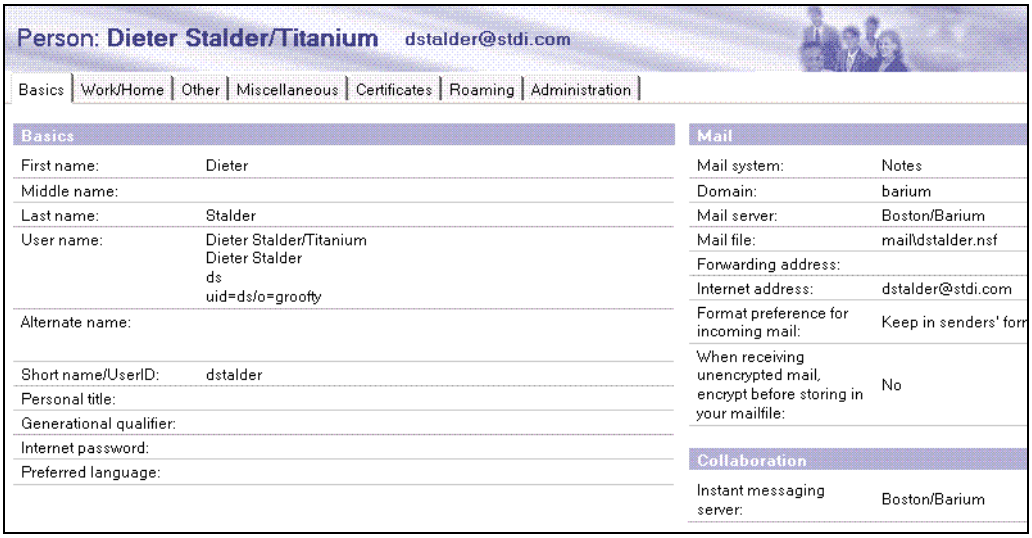


Figure 6-12 Enter the LTPA user name in the User name field

Internet Site: LDAP

If the LDAP configuration does not exist, select **Add Internet Site** → **LDAP**.

On the Basics tab, provide values for the following fields, as shown in Figure 6-13:

- Descriptive name for this site

  Enter a description for the site.

- Organization

  The organization name for LDAP document has to match the name of the certified organization. In our test setup, the server name is Toronto/Titanium, so the organization name is titanium.

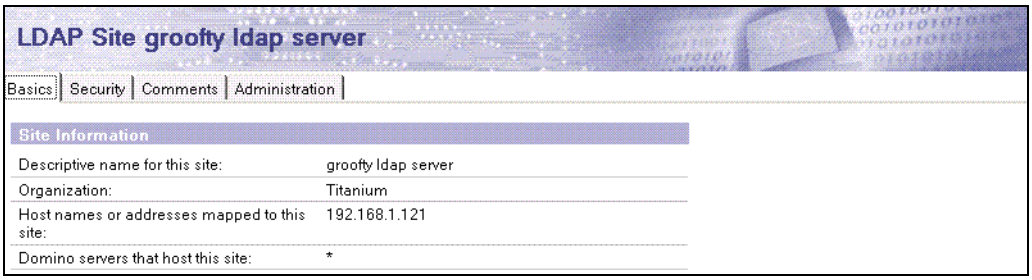


Figure 6-13 Adding an LDAP Site document

Under the Security tab, provide values for the following fields, as shown in Figure 6-14:

- TCP Authentication

  By default, Anonymous access is enabled. Change Anonymous to **No**.

- SSL Authentication

  By default, Anonymous access is enabled. Change Anonymous to **No**.

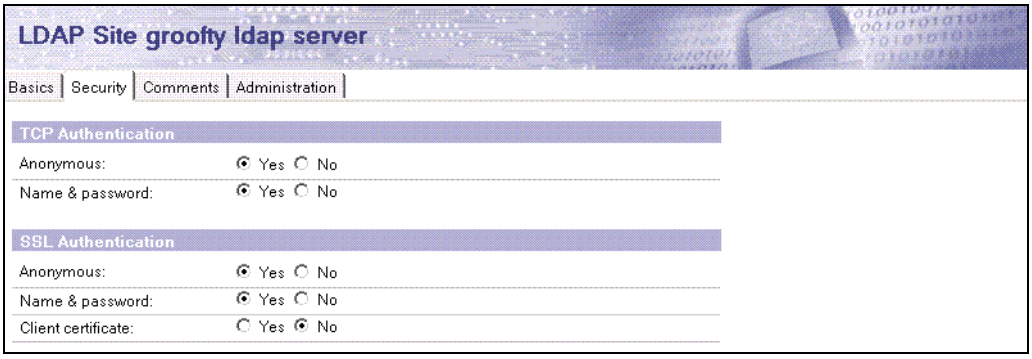


Figure 6-14 Security tab in the LDAP Site document

6.1.3 LDAP directory assistance configuration (gateway)

This section describes the directory assistance configuratoin.

SSO configuration

Select **Directory Assistance** → **Basic** → **SSO Configuration**. The Directory Assistance document in R7 has a new field to set the LDAP attribute name for the LTPA user name, as shown in Figure 6-15 on page 89.

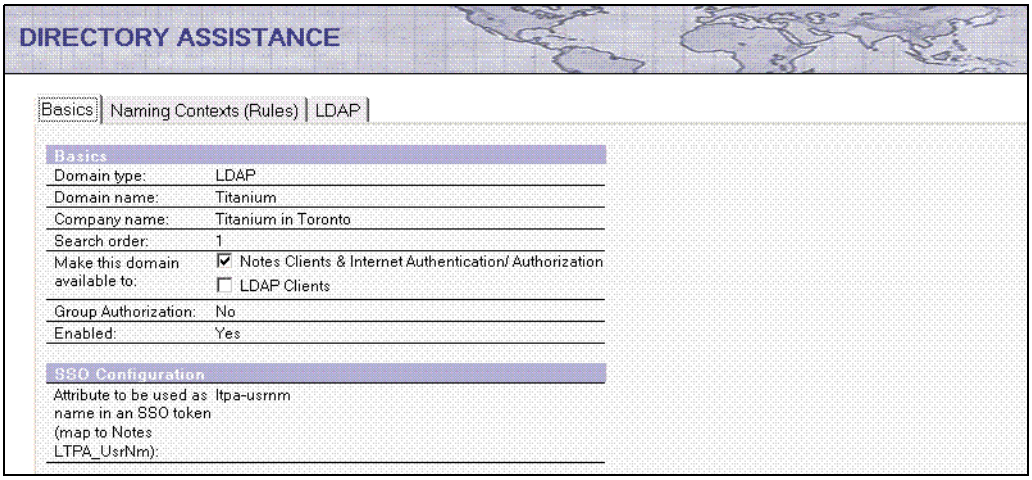


Figure 6-15 Setting the LDAP attribute name

Note: The LTPA user name in the Domino Directory is LTPA_UsrNm (with underscore), and the LDAP attribute name is LTPA-UsrNm (with hyphen). Directory assistance accesses the LDAP attribute and the field name is therefore LTPA-UsrNm.

The setting in directory assistance controls the LTPA user name from the LDAP directory, not the local Domino Directory. The local Domino Directory was configured in the first step, in “Web SSO configuration” on page 81.

Naming Context (Rules) configuration

Select **Directory Assistance** → **Naming Context (Rules)**. In the Naming Contexts (Rules) tab, select **Yes** in the Trusted for Credentials column, as shown in Figure 6-16. This is required for authentication.

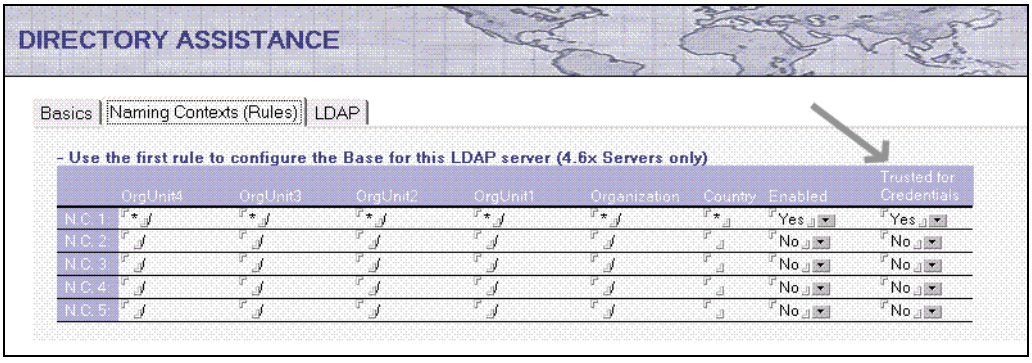


Figure 6-16 Setting the value in the Trusted for Credentials column

Attribute for Notes configuration

Select **Directory Assistance** → **LDAP** → **Attribute for Notes**. This is where the configuration gets a bit more complex. Up to now, we changed the selection of check boxes and entered names. Now, we have to teach Domino that the name in the LTPA token is not the one to use for access control and the @UserName function.

When the Domino server authenticates using the user name in the LTPA token, the user is identified as uid=ds,o=groofty. Keep in mind that this user does not have a Person document

in the current Domino Directory. Authentication is based on the directory assistance LDAP. Unless we tell Domino that this is not the name to use for security, the server uses the name from the LTPA token. For example, a database has documents with Read-Access fields. These fields store the user name as cn=Dieter Stalder/o=Titanium. Domino compares the name from the LTPA token (uid=ds) with the Notes name (cn=Dieter) and there is no match. Therefore, we do not have access. We have two options to solve the problem:

1. Use the foreign name. Add uid=ds/o=groofty to the Reader field and Domino will authenticate correctly and grant access. Although this is not a recommended solution, it works.
2. Tell Domino to translate the foreign name into the Notes name. This is the reason we configure the attribute that returns the Notes distinguished name (DN).

We need to change the LDAP server configuration, but first we add the LDAP attribute name to directory assistance, as shown in Figure 6-17. The attribute name is, or will be `owner`. Save the Directory Assistance document in the Domino application server and change to the Domino LDAP server.

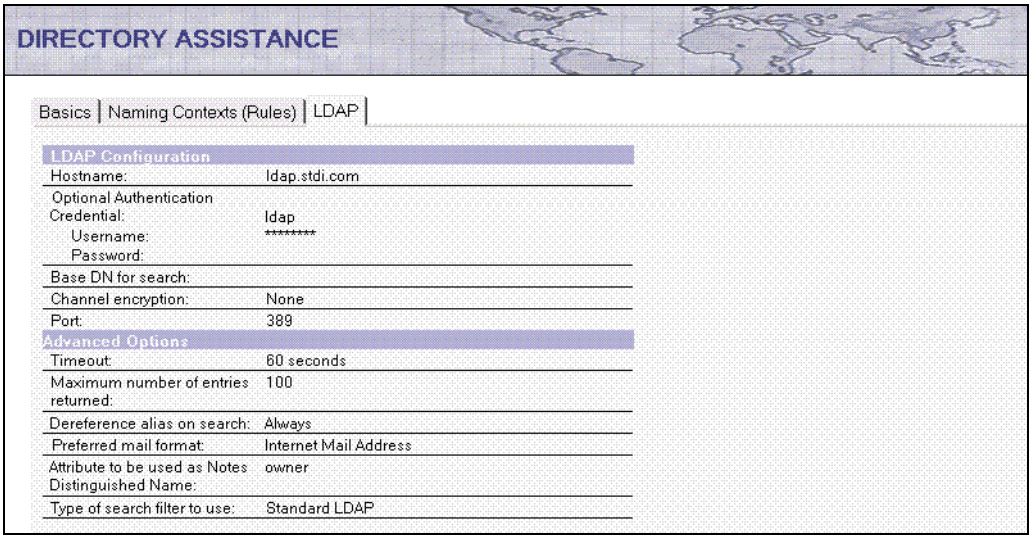


Figure 6-17 Add the LDAP attribute name to the Directory Assistance document

Person document changes with agent

In the Person document, we need to add a field that stores the user's canonical name (cn=Dieter Stalder/o=Titanium). There is no existing field to use, so we have to create one. To avoid design changes in the Person document form, we use the Owners field on the Administration tab, and then run an agent to copy this field into one that can be configured in LDAP. See Figure 6-18.

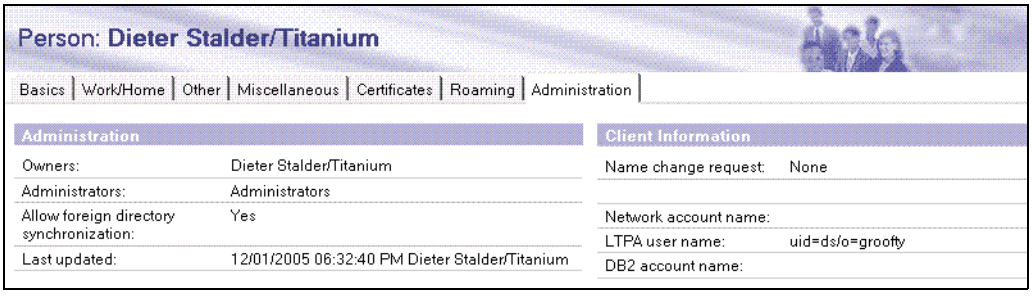


Figure 6-18 Changes to the Person document

The Owner field should have the first occurrence from the User name field. The Owner field is an Authors field; when saving the Person document, the name is stored in canonical format (cn=Dieter Stalder/0=Titanium).

Next, we copy the value from the Owner field in the Person document to the LDAPOwner field. In our example, we create an agent with the copy statement.

In the Domino Directory, switch to the People view. Then select **Create** → **Agent** from the menu.

Then select all the Person documents and run the agent. Now we have the Notes canonical name in the LDAPOwner field. See Figure 6-19.

Note: The value in the LDAPOwner field is in the Notes format (with slashes). The owner attribute is in LDAP format (with comma).

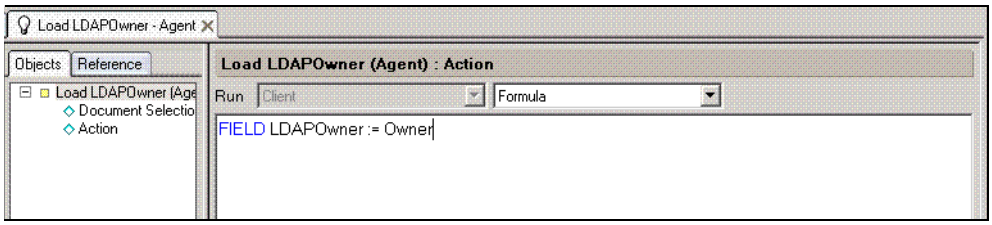


Figure 6-19 Creating the agent

Configuration document: LDAP

We now configure the mapping of the LDAP attribute owner and the Notes field *LDAPOwner*. Perform the following steps:

1. Open the default Configuration document *** = [All Servers]**.
2. Go to the LDAP tab.
3. Click **Select Attribute Types**.
4. Click **New** at the bottom window, as shown in Figure 6-20 on page 92.

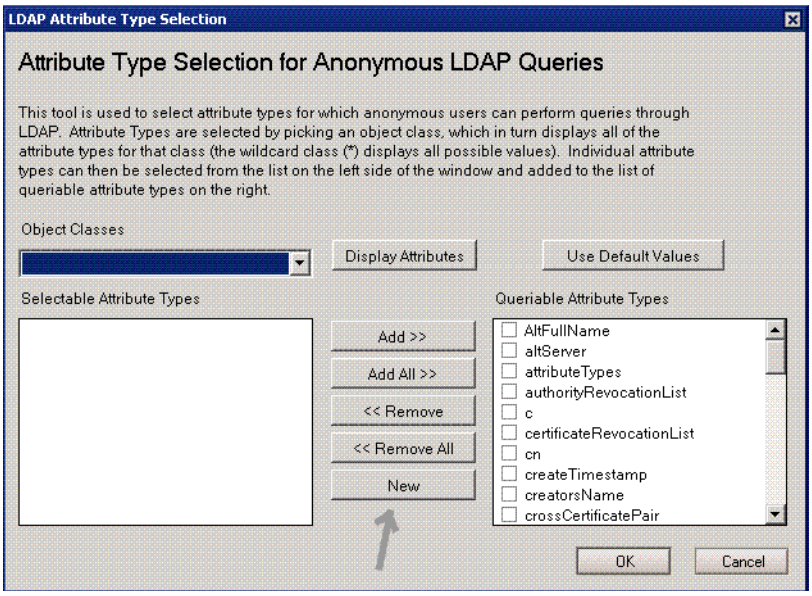


Figure 6-20 Configure the mapping of the LDAP attribute

5. Enter `owner` in the New Field window.

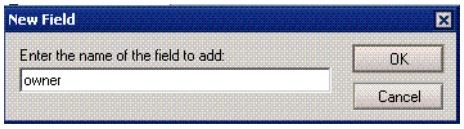


Figure 6-21 Entering owner

6. Click **OK**.

   The list of field names now includes the new name owner in the LDAP Attribute Type column and LDAPOwner in the Domino fields.

7. Save the changes.
8. Restart the LDAP server.

6.1.4 Verify LDAP with the ldapsearch utility

Verify the configuration with the **`ldapsearch`** utility. The utility is in the \Lotus\Notes or \Lotus\Domino folder. Enter the search key for a person and **`ldapsearch`** returns all the attributes, as shown in Example 6-1.

Example 6-1 Results from searching for a person in LDAP search

```
ldapsearch -h ldap.stdi.com "cn=dieter stalder"
CN=Dieter Stalder,O=Titanium
cn=Dieter Stalder
cn=ds
mail=DieterStalder@stdi.com
owner=CN=Dieter Stalder,O=Titanium
usercertificate;binary=NOT ASCII
objectclass=dominoPerson
objectclass=inetOrgPerson
objectclass=organizationalPerson
objectclass=person
objectclass=top
```

```
dominocertificate=03009702 9A0DC617 0AG0161C G002E3F9
ltpa-usrnm=uid=DieterStalder,o=groofty
givenname=Dieter
sn=Stalder
uid=DStalder
maildomain=Titanium
```

Verify that LDAP returns the user information based on the foreign user ID, as shown in Example 6-2.

Example 6-2 User information based on the foreign user ID

```
ldapsearch -h ldap.stdi.com -a always -b "uid=ds,o=groofty" objectclass=*
CN=Dieter Stalder,O=Titanium
cn=Dieter Stalder
cn=ds
mail=dstalder@stdi.com
owner=CN=Dieter Stalder,O=Titanium
...
```

Refer to the “ldapsearch utility” topic in the Lotus Domino 7 Administrator Help database for a complete description of the parameters, available at:

http://www.lotus.com/ldd/doc/domino_notes/7.0/help7_admin.nsf/

6.2 Considerations and examples

In this section, we provide some general guidance and examples.

Note: Make sure that all system clocks are synchronized. The LTPA token includes a creation and expiration time. When the time expires, Domino prompts for the user ID and password again. We encountered this issue. After synchronizing the time zone and time on the servers, the results started to make sense. The error messages did not indicate that the time expired.

6.2.1 Upgrading from previous versions

The Domino Web configuration changed over the last few releases. Feature backward compatibility might become an issue when upgrading to R7. To take full advantage of SSO and Web site configuration, now is a good time to switch to Internet Site documents, introduced in R6, and now with R7, also compatible with QuickPlace.

Converting Web site redirection to Internet site rules

If you configured redirection documents under the Web server configuration, you need to re-create them as Web Site Rules for the Internet Site documents. If you have more then a few documents, this conversion process can be time consuming and error prone.

The agent code shown in Example 6-3 on page 94 enables you to select the redirection document from the Web server configuration and copy the From and To strings into the new Internet Site Rules document. To create the agent, perform the following steps:

1. Open the Domino Directory and select **Create** → **Agent**.
2. Give the agent a name.
3. Select **Private** from Options.

4. Change the Runtime Target to **None**.
5. Then, copy the code in Example 6-3 to the agent.

Example 6-3 Agent code enables you to select the redirection document

```
@All ;
@Command( [EditGotoField] ; "MappingType"  ) ;
@Command( [EditSelectAll] ) ;
@Command( [EditInsertText] ; "Redirection" ) ;
t1 := @PickList([Custom];"" ; "WebConfigurations" ; "Old" ; "Pick" ;  2) ;

in1 := @Right(t1 ; "(" ) ;
in2 := @Left(in1; " " ) ;
@Command( [EditGotoField] ; "RM_MapFrom"  ) ;
@Command( [EditSelectAll] ) ;
@Command( [EditInsertText] ; @Trim(in2) ) ;

ou1 := @RightBack(t1 ; " " ) ;
ou2 := @Left(ou1 ; ")" ) ;
@Command( [EditGotoField] ; "RM_MapTo"  ) ;
@Command( [EditSelectAll] ) ;
@Command( [EditInsertText] ; @Trim(ou2) ) ;
```

6.2.2 Enabling SSL for SSO

If SSL is not already turned on, now is a good time to do so. An unencrypted SSO cookie can be captured and reused.

6.2.3 Lotus QuickPlace 7: Installation notes

The QuickPlace test setup uses two servers in different Domino domains, as shown in Figure 6-22. One is the LDAP server where all users are configured, and the other is the QuickPlace server. Users are configured in the LDAP directory. The servers are configured with SSO and LTPA token name mapping.

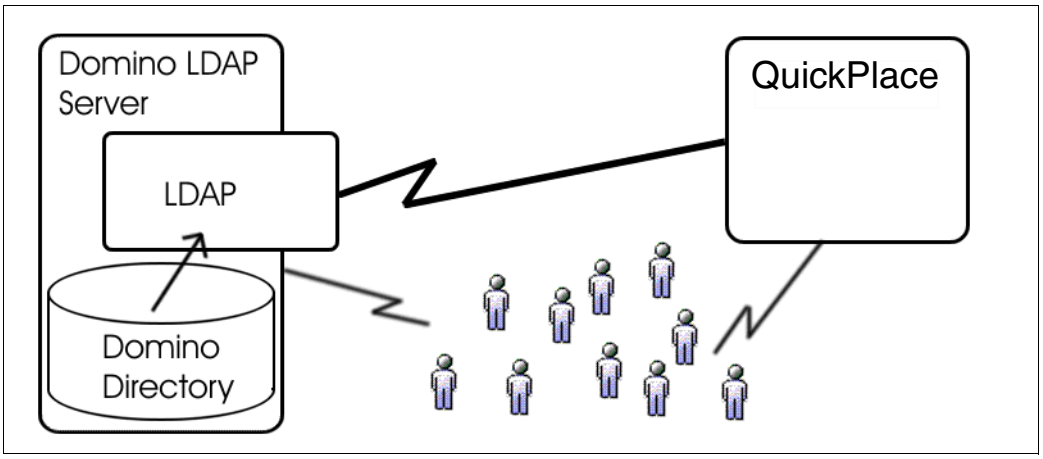


Figure 6-22 Overview of the test SSO environment with QuickPlace

QuickPlace 7 accepts Internet Site documents.

Authentication worked correctly and the LTPA token name was correctly set. Logging in to the LDAP server first or to the QuickPlace server first produced the same results.

Some documentation refers to the QuickPlaceRemapDN= NOTES.INI parameter. We tested without the parameter and authentication worked correctly. For complete QuickPlace installation instructions, refer to the QuickPlace documentation.

Figure 6-23 shows the QuickPlace (formerly called Lotus Team Workplace) welcome window.

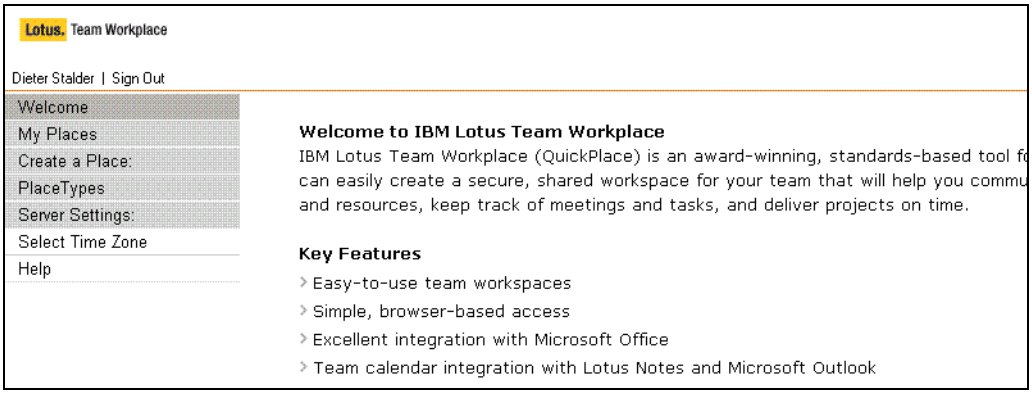


Figure 6-23 QuickPlace welcome window

6.2.4 Lotus Sametime 7: Installation notes

The Sametime test setup uses two servers in different Domino domains, as shown in Figure 6-24. One is the LDAP server where all users are configured, and the other is the Sametime server. Sametime users are configured in the LDAP directory. The servers are configured with SSO and LTPA token name mapping.

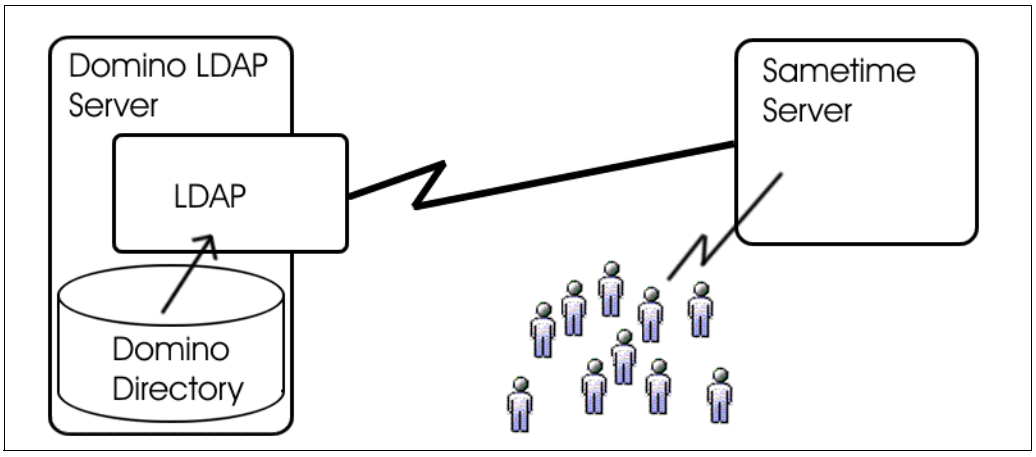


Figure 6-24 Overview of the test SSO environment with Sametime

Sametime expects Web Configuration documents. If the Web SSO Configuration is missing, the installation process creates one. When prompted to setup HTTP tunneling, the installation process modified the Server document and changed HTTP port 80 to 8088.

Authentication worked correctly and the LTPA token name was correctly set. Logging in to the LDAP server first or to the Sametime server first produced the same results.

The installation process did not allow for an LDAP configuration with authentication credentials. After the installation, adjustments were made in the LDAP Server document in the Sametime Configuration database. The Login Name for LDAP Connection and Password for LDAP Connection were added.

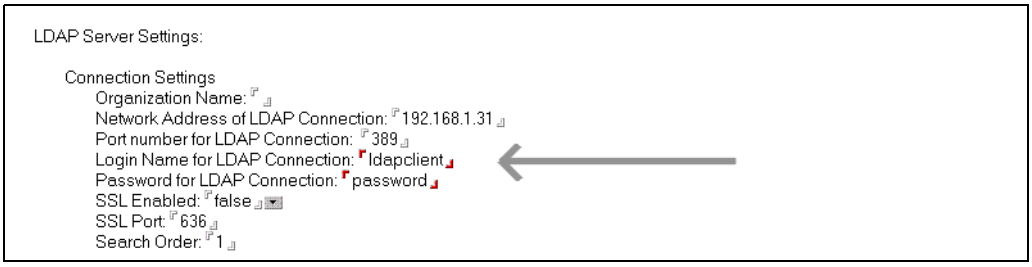


Figure 6-25 Login Name for LDAP Connection and Password for LDAP Connection

Directory assistance

The install process adds an LDAP entry to the directory assistance database (or creates directory assistance). Check the server log file for errors. The entry might be missing the Authentication Credentials or the LDAP entry might be duplicated. If you already had an existing entry, the Sametime installation adds a second entry.

NOTES.INI

Sametime has to know how to read the LTPA token name and translate it to the Notes name. See Example 6-4.

Example 6-4 NOTES.INI file: Setting LTPA token name for translation to the Notes name

```
ST_UID_PREFIX=uid=
ST_UID_POSTFIX=,
```

The LTPA token user name is uid=ds,o=groofty. The ST_UID_PREFIX defines that the user name is after the uid= string, and the user name continues until a comma is found, as per the ST_UID_POSTFIX setting. In this example, the user name ds has to be a unique identification for the user.

The second change is in the SAMETIME.INI file, in the [Directory] section. See Example 6-5. This parameter is documented for Sametime 6.5.1. Testing with Sametime 7 worked without this parameter.

Example 6-5 Changes in the SAMETIME.INI file

```
[Directory]
ST_DB_LDAP_DEREF=3
```

For more details about the Sametime installation, refer to the *Sametime Installation Guide* and *Release Notes*.

Figure 6-26 on page 97 shows the Sametime login window.

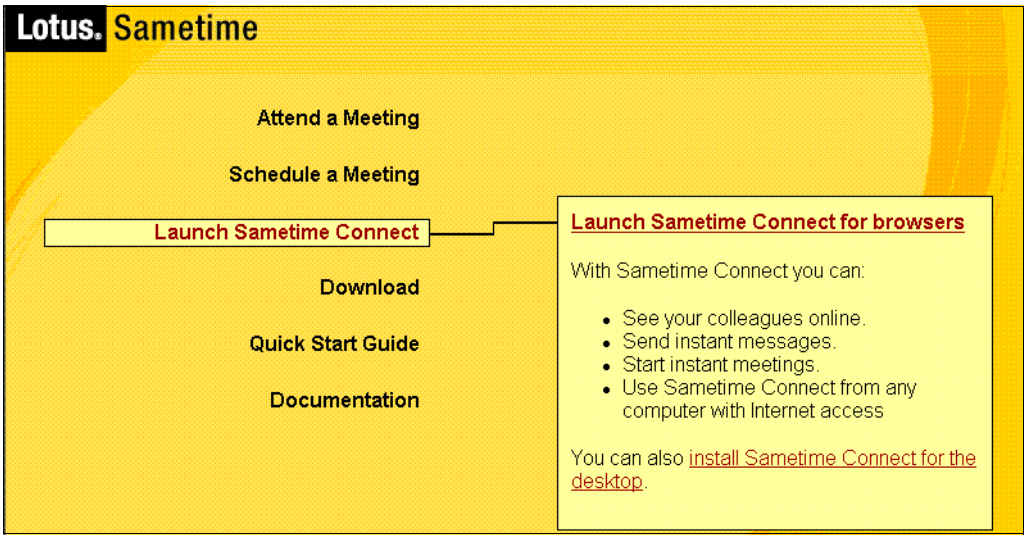


Figure 6-26 Sametime login window

6.2.5 SSO debug instructions

When setting up SSO, you can verify the LTPA user name by enabling SSO debugging. Every connection creates lots of debugging information. Use the setting with caution. The user name in the LTPA token is in LDAP format (with comma). Domino correctly converts between the internal format (with slash) and the LDAP format (with comma).

Set the `DEBUG_SSO_TRACE_LEVEL=2` parameter in the NOTES.INI file on the Domino server.

Example 6-6 shows the organization and token name in the first data block. The second data block shows the time stamps and user name. In this example, the token has the LTPA user name from the Person document (`uid=DieterStalder`).

Example 6-6 Detailed output

```
[0788:0013-0494] SSO API> Dumping memory of encoded token [92 bytes].
00000000: 4141 4345 7A41 7A51 6B4E 3249 4452 4552   'AAECAzQzNkI2RDRE'
00000010: 444E 324D 6A51 3063 544E 3156 5761 3951   'NDM2QjcONTV1aWQ9'
00000020: 4752 6C6C 4764 7956 3355 6852 4762 6C52   'RGlldGVyU3RhbGRl'
00000030: 6963 7639 5750 7964 3262 6D39 4864 7A6C   'ci9vPWdyb29mdHlz'
00000040: 4250 6F43 574F 5958 6A49 5776 4C5A 4141   'PBCoOWXYIjvWZLAA'
00000050: 326C 3275 3649 3570 7759 3D3D             'l2u2I6p5Yw=='
[0788:0013-0494] SSO API> *** Retrieving Extra Token Info (SECTokenValidateAndGetTokenInfo)
***
[0788:0013-0494] SSO API> OrgName specified [QP].
[0788:0013-0494] SSO API> ConfigName specified [LtpaToken].
[0788:0013-0494] SSO API> Retrieved global static cache memory for config [QP:LtpaToken].
[0788:0013-0494] SSO API> Decoding Domino style Single Sign-On token.
[0788:0013-0494] SSO API> Dumping memory of encoded token [92 bytes].
00000000: 4141 4345 7A41 7A51 6B4E 3249 4452 4552   'AAECAzQzNkI2RDRE'
00000010: 444E 324D 6A51 3063 544E 3156 5761 3951   'NDM2QjcONTV1aWQ9'
00000020: 4752 6C6C 4764 7956 3355 6852 4762 6C52   'RGlldGVyU3RhbGRl'
00000030: 6963 7639 5750 7964 3262 6D39 4864 7A6C   'ci9vPWdyb29mdHlz'
00000040: 4250 6F43 574F 5958 6A49 5776 4C5A 4141   'PBCoOWXYIjvWZLAA'
00000050: 326C 3275 3649 3570 7759 3D3D             'l2u2I6p5Yw=='
[0788:0013-0494] SSO API> Dumping memory of decoded token [67 bytes].
00000000: 0100 0302 3334 4236 4436 4434 3334 4236   '....436B6D4D436B'
00000010: 3437 3535 6975 3D64 6944 7465 7265 7453   '7455uid=DieterSt'
```

```
00000020: 6C61 6564 2F72 3D6F 7267 6F6F 7466 7379   'alder,o=grooftys'
00000030: 103C 39A8 D865 3B22 64D6 00B0 6B97 23B6   '<.(9eX";Vd0..k6#'
00000040: 79AA   63                                 '*yc'
[0788:0013-0494] SSO API> -Creation Ticks   = 436B6D4D [11/04/2005 09:16:45 AM].
[0788:0013-0494] SSO API> -Expiration Ticks = 436B7455 [11/04/2005 09:46:45 AM].
[0788:0013-0494] SSO API> -Username         = uid=DieterStalder,o=groofty
```

Example 6-7 shows the Domino user name as defined in the first position of the user name in the Person document (`CN=Dieter Stalder`).

Example 6-7 Domino user name as defined in the first position of the User name

```
[084C:0013-00F4] SSO API> Dumping memory of decoded token [68 bytes].
00000000: 0100 0302 3334 4236 4536 3341 3334 4236   '....436B6EA3436B'
00000010: 3537 4241 4E43 443D 6569 6574 2072 7453   '75ABCN=Dieter St'
00000020: 6C61 6564 2F72 3D4F 6954 6174 696E 6D75   'alder,O=Titanium'
00000030: A9BD EB3C 4E7B 11BD 35CD 0BC9 AD4B C1C0   '=)<k{N=.M5I.K-@A'
00000040: 9A29 3D59                                 ').Y='
[084C:0013-00F4] SSO API> -Creation Ticks   = 436B6EA3 [11/04/2005 09:22:27 AM].
[084C:0013-00F4] SSO API> -Expiration Ticks = 436B75AB [11/04/2005 09:52:27 AM].
[084C:0013-00F4] SSO API> -Username         = CN=Dieter Stalder,O=Titanium
```

7

Securing Domino Web Access

Lotus Domino Web Access is a Web client that enables users to access different Domino services using a Web browser. It provides the browser user with access to a number of advanced client-side features and functionality that were generally only available for users with non-browser clients, such as Lotus Notes. These advanced features greatly enhance the client experience in the areas of messaging, calendar and scheduling, personal information management (PIM), task management, and personal journal. Users can also work offline to manage e-mail messages, contacts, calendars, to-do items, and so forth from the user interface that Domino Web Access provides.

Domino Web Access has now been available for a number of years now and has evolved with each new release of the Domino server. For Release 7.0, this is no exception. The parity with Lotus Notes in terms of functionality is rather close in the previously listed areas, including Lotus Sametime integration. However, the Notes client still provides a richer, more complete end-user experience, especially when it comes to the design, use and replication of applications, the support of PDA devices, and some other features available only with this client.

Domino Web Access was originally designed to provide a browser with most of the features of Notes. It was also designed for a type of user called the *diskless user*. This type of user typically is one that requires only occasional access to e-mail and a calendar. Such users are mobile and do not have a fixed location, making them ideally suited for the type of functionality offered by a browser and Domino Web Access. The same goes for external users that collaborate closely with the organization and require access to the organization's messaging system, but for whom the organization does not want to provide a Notes client.

However, a new class of users has emerged that makes full use of Domino Web Access, and those are users who would normally receive Notes client, but who are generally remote and for which the cost of implementing a Domino server at the location, implementing individual Notes clients, and providing on-site support would be too expensive. So, these users are generally given Domino Web Access, which provides most of the functionality with considerable savings.

With all this in mind, in this chapter, we discuss the new security functionalities offered by Domino 7 that enhance the security of Domino Web Access. In the previous security redbook, *Lotus Security Handbook*, SG24-7017, the information pertaining to Domino Web Access is

under iNotes™, the previous name of Domino Web Access, in Chapter 12, “Security features of other Lotus products.”

We cover some necessary basics to ensure that Domino Web Access is properly installed and works as expected, providing some best practices along the way. In addition, because this is a security book, we cover the following security functionalities and enhancements:

- Browser Cache Management: This feature improves client performance and security of Domino Web Access sessions on Microsoft Internet Explorer, because it controls which entries are stored in the cache and which are removed when the Domino Web Access session ends.
- S/MIME support: Secure/Multimedia Internet Mail Extensions (S/MIME) is now supported in Domino Web Access, which permits the exchange of secure messages over a Web browser in conjunction with Domino Web Access.

Because the features of Domino Web Access exist as intersection points between the Web browser and the Domino server, and are generally a source of confusion leading to less-than-optimal security configurations, we describe where the specific security feature and functionality is applied, be it on the Domino server, in the Web browser, or both.

Finally, it must be understood which platforms are supported to ensure that Domino Web Access works properly and to avoid any problems later.

Domino Web Access 7.0 supports the following server operating systems:

- Microsoft Windows 2000 Server SP4
- Microsoft Windows 2000 Advanced Server SP4
- Microsoft Windows 2003 Standard Edition
- Microsoft Windows 2003 Advanced Edition
- IBM AIX® 5L™ Version 5.2
- IBM AIX 5L Version 5.3
- Sun™ Solaris™ 9 (NSF only)
- IBM i5/OS® V5R3 (NSF only) for IBM @server® iSeries™
- IBM z/OS 1.5 (NSF only) for IBM @server zSeries®
- Novell SUSE Linux® Enterprise Server (SLES) 8 SP3 (NSF only) for IBM @server zSeries
- Novell SUSE Linux SLES 9 SP1 (NSF only) for IBM @server zSeries
- Novell SUSE Linux SLES 8 SP3 (NSF only) for Intel®
- Novell SUSE Linux SLES 9 SP1 (NSF only) for Intel

Domino Web Access 7.0 supports the following client operating systems:

- Microsoft Windows 2000 Professional SP4
- Microsoft Windows XP SP2 (Professional only)
- Novell Linux Desktop (NLD) 8
- Novell Linux Desktop (NLD) 9

Domino Web Access 7.0 supports the following Web browsers:

- Microsoft Internet Explorer 6.0 on Windows 2000 and Windows XP
- Mozilla Browser 1.7.x (Linux only)

- Mozilla Firefox 1.0.4 Browser on Windows 2000, Windows XP, and Linux 7.x

In addition, in Version 7.0.2, there will be support for Mac clients using the Firefox browser.

7.1 Overview of Domino Web Access

From a technical perspective, Domino Web Access (previously iNotes Web Access) provides Lotus Notes users with browser-based access to Notes mail and to Notes calendaring and scheduling features. Domino Web Access users can send and receive mail, view their calendars, invite people to meetings, create to do lists, keep a notebook, and work offline.

After being set up for Domino Web Access, a user can use both the standard Notes client and a Web browser to access their mail files. Because both the Notes client and Domino Web Access operate on the same underlying user mail file, read and unread marks remain up-to-date, regardless of which client the user uses to read the mail. Users can also synchronize contact information in their Personal Address Book with information in their Contact List in Domino Web Access.

In addition, to provide users with the ability to both work offline and use Sametime. To work offline, it is possible to integrate Domino Web Access with Domino Off-Line Services. Domino Off-Line Services enables users to work offline, disconnected from the network, and provides many replication features that Notes users expect when working in the Notes client. It is also possible to integrate Domino Web Access with Lotus Sametime to provide integrated, real-time chat features for Domino Web Access users. It is worth noting that neither Domino Off-Line Services nor Sametime are required for Domino Web Access use.

In terms of security, there are a few basic things to remember when it comes to Domino Web Access.

First, Domino Web Access requires user logon security (logout is not required although it is a good practice). When users log on to Domino Web Access, they must enter their name and Internet password, as specified in their Person document. The login names that the server accepts as valid depend on the setting in the “Internet authentication” field on the Security tab of the Server document.

Second, while users simply need a name and Internet password to log on and use Domino Web Access, a Notes ID is required for using secure mail. Administrators need to ensure that they create a Notes ID for each user when registering new users with the Domino Web Access template.

We cover the more advanced security concepts in further detail in the rest of this chapter.

7.2 Setting up Domino Web Access

Before we start discussing the security specifics of Domino Web Access, it is important to take a moment to ensure that Domino Web Access is properly setup and working on the Domino server.

It is not our intention to cover all aspects of configuring the Domino server and Domino Web Access. Refer to the following IBM Redbooks for details about this topic (which apply to Releases 6.5 and 5.0.9, respectively):

- *Domino Web Access 6.5 on Linux*, SG24-7060, available at:

  http://www.redbooks.ibm.com/abstracts/sg247060.html

- *iNotes Web Access Deployment and Administration*, SG24-6518, available at:

  http://www.redbooks.ibm.com/abstracts/sg246518.html

For this book, note the following three items:

- First, it is possible to set up three different types of servers in Version 7:
  - Domino Utility Server
  - Domino Messaging Server
  - Domino Enterprise Server

  Of these three, only the Domino Utility Server should not be selected during the installation, because it is a new installation type for Domino Release 7.0 that does *not* include support for messaging services.
- Second, during the configuration of the server (which occurs the first time the server is launched after a new installation), it is possible to select the following Internet services the server should provide: Web browsers (HTTP services), Internet mail clients (SMTP, POP3, and IMAP services), and Directory services (LDAP services). At a minimum, select Web browsers (HTTP services) to ensure that the HTTP task is loaded, which is a basic requirement for Domino Web Access.
- Third, as shown in Figure 7-1, there are three mail templates that ship out of the box with the Domino 7.0 serve: the Domino Web Access (7) template (dwa7.ntf), Extended Mail (R7) template (mail7ex.ntf), Mail (R7) template (mail7.ntf), and Domino Web Access (6) template (iNotes6.ntf).

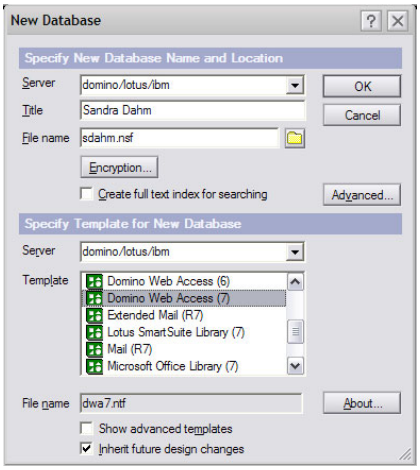


Figure 7-1 The mail templates that ship out of the box with Domino 7

New users need to be registered with the dwa7.ntf template. Existing users mail files (if not registered using the dwa7.ntf template) need to have the design of their mail database replaced with the dwa7.ntf template. This is the only template that contains mail template support for the Domino Web Access client and the Notes client. Figure 7-2 on page 103 illustrates what the new interface looks like.

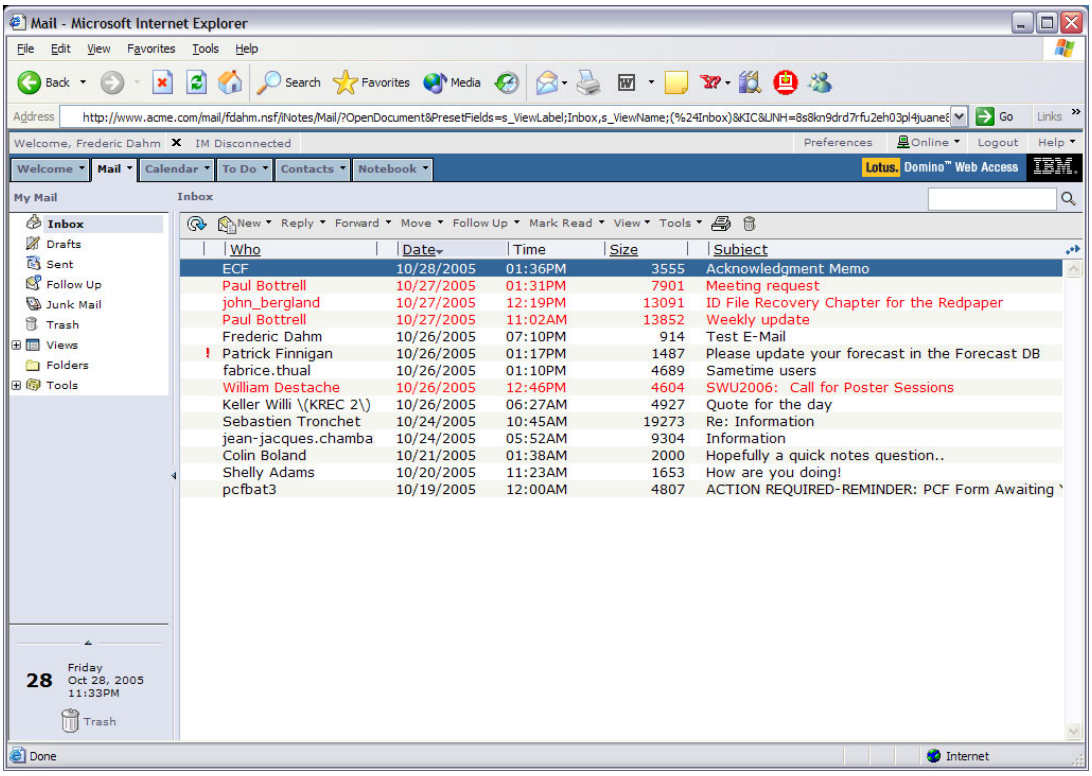


Figure 7-2 The new look of Domino Web Access

Note: The mail7.ntf template is the standard template. On a server running the HTTP task, if opened in a Web browser, the user obtains the Web Mail interface (a simple HTTP interface, supplemented with some Java™ applets, for the user's mail database). Use the Extended Mail Template (Mail7ex.ntf) for all mail files migrated from Microsoft Exchange to Domino. In addition, while the dwa7.ntf template has the new look of the Domino Web Access interface, the iNotes6.ntf template uses the old interface. All in all, for the purpose of this chapter, ensure that you use the dwa7.ntf template.

Now that we have properly configured the server, the Internet services it provides, and ensured that the e-mail design is based on the proper template, we are now ready to address the topic of authentication and Domino Web Access.

7.3 Domino Web Access authentication

Domino Web Access is no different from other Domino databases residing on the server in that if the access control list (ACL) is not configured properly, the database might be accessible by unauthorized third parties. If, for example, Anonymous is set in the ACL to a level of author and above, someone could use a Domino Web Access database to send e-mails. However, the e-mail will not be send in the name of the owner of the database, but as Anonymous and received as such in the recipients mailbox, as shown in Figure 7-3 on page 104.

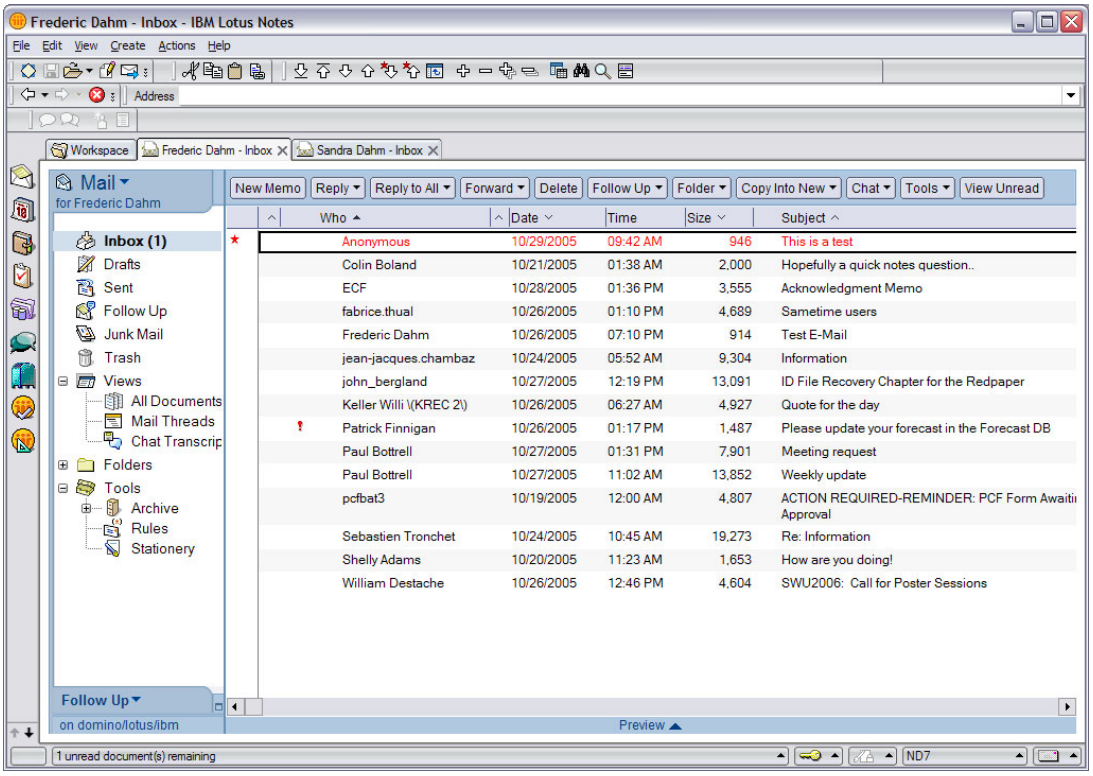


Figure 7-3 E-mail received from user Anonymous

Therefore, effective security requires that Domino Web Access users authenticate themselves. In other words, the ACL of the user's mail file must be set properly to force users to log on and log out in order to use the features and functionality of Domino Web Access.

When a user logs on to Domino Web Access, that person must enter a user name and the corresponding Internet password, as specified in their Person document. The login names that the server accepts as valid depend on the setting in the "Internet authentication" field on the Security tab of the Server document.

What can vary is the manner in which authentication is carried out. For Domino Web Access, users can authenticate in a number of ways, depending on the configuration of the authentication mechanisms on the Domino server:

- Simple user ID and password authentication, through the Web browser's authentication dialog box.
- Session authentication, through a login form. With session authentication, this can also include single sign-on (SSO), which ensures that the user logs in once (for example, at the organization's portal) and does not need to log in again and again when accessing different servers that are part of the single sign-on domain.
- Certificate-based authentication using x.509v3 client certificates.

It is important to note that in an effort to ensure optimal security, the communication channel can be encrypted between the user's Web browser and Domino using Secure Sockets Layer (SSL).

We quickly cover the specifics of what each authentication mechanism involves and some variants that can be applied in order to extend the authentication mechanism.

Simple name and password authentication

This type of authentication is configured by default when the Domino server is first setup. It is triggered when the user tries to access a database whose ACL denies access to anonymous users and requires identification to grant an appropriate level of access. Figure 7-4 shows the dialog box that the Web browser displays when authentication is required by the Domino server.

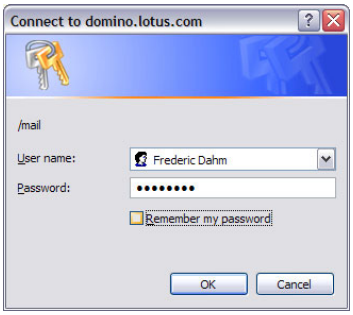


Figure 7-4 Simple name and password authentication

Session authentication

This type of authentication is configured through the Server document, through the Internet Protocols tab and the Domino Web Engine subtab by setting the Session authentication from disabled (for simple name and password authentication) to **Single Server**, as shown in Figure 7-5.

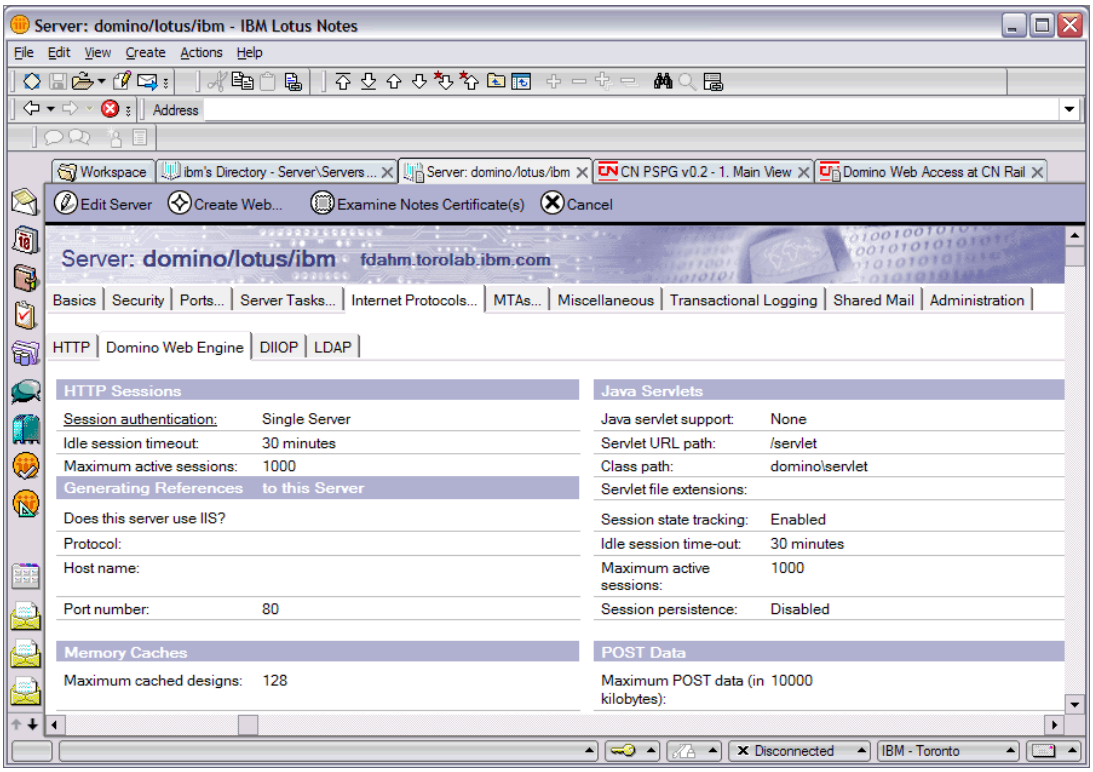


Figure 7-5 Setting Session authentication to Single Server

Note that the HTTP task must be restarted for the parameter to take effect. Do this by invoking the **`tell http restart console`** command. When the user authenticates, the Server Login form opens, as shown in Figure 7-6 on page 106, instead of the dialog box.

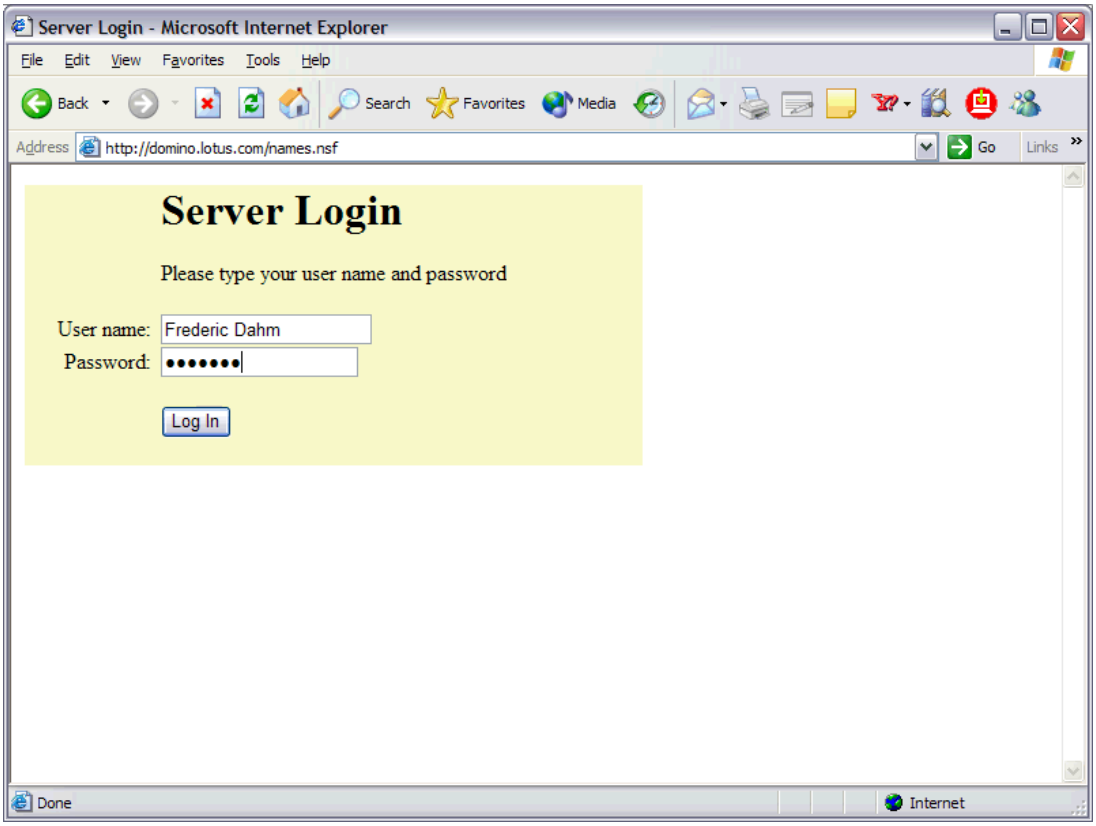


Figure 7-6 The basic Server Login form

User might find this page drab and unprofessional looking. To improve the look of this page, create the Domino Web Server Configuration database (DOMCFG.NSF), as shown in Figure 7-7.

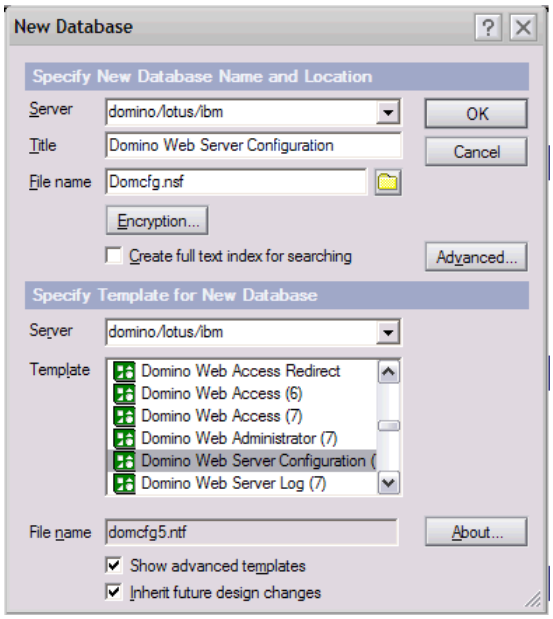


Figure 7-7 Creating the Domino Web Server Configuration database

Figure 7-8 shows the page after creating the database and restarting the HTTP.

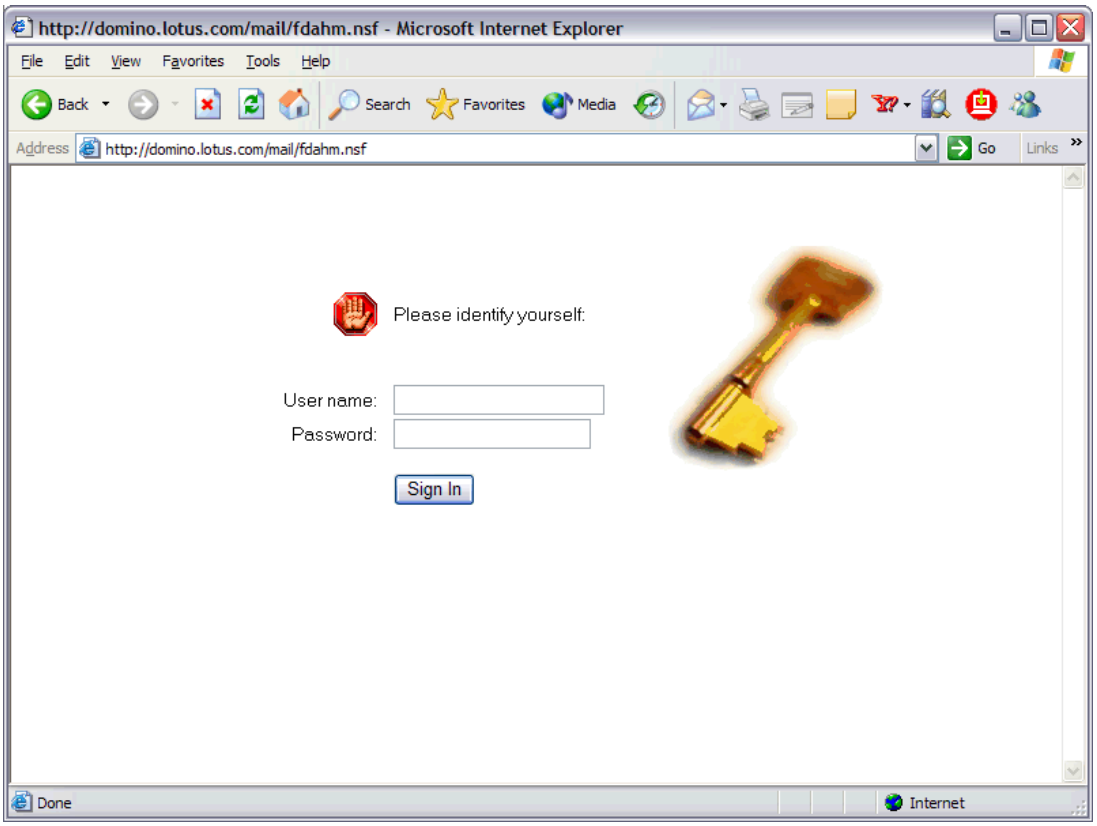


Figure 7-8 An improved server login window

In addition, you can further improve to give the full Domino Web Access experience to Domino Web Access users.

Set up Domino Web Access Redirect using the Domino Web Access Redirect template (IWAREDIR.NTF), which is in the server's Domino data directory. To use the new DWALoginForm, which provides a full Domino Web Access experience to end users:

1. Open the Domino Web Server Configuration database (DOMCFG.NSF).
2. Click **Add Mapping**.
3. Change the Target Database to your Domino Web Access Redirect database.
4. Change the Target Form to **DWALoginForm**.
5. Click **Save & Close**.

After completing these steps and restarting the HTTP task, you can use the new DWALoginForm, which will look like the one shown in Figure 7-9 on page 108.

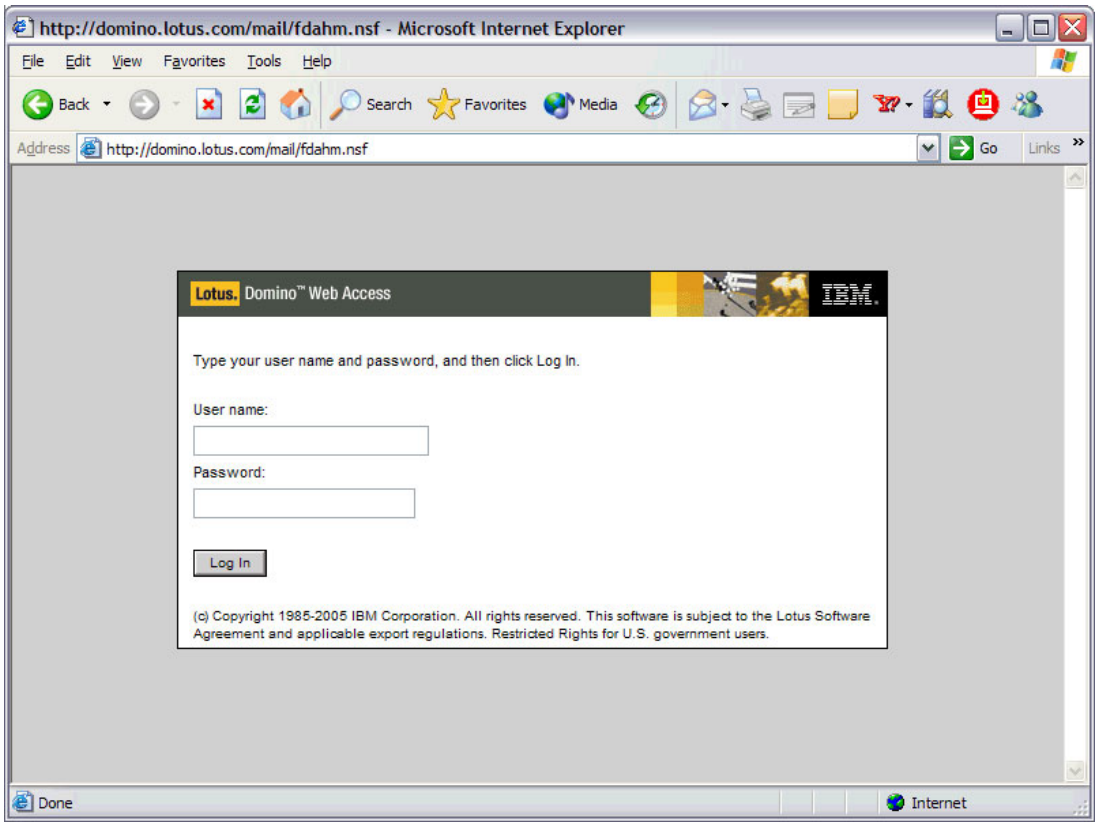


Figure 7-9 The DWALoginForm server login form

With session authentication, it is also possible to extend the authentication services in a couple of ways.

First, you can include something such as RSA Ace/Agent and RSA SecurID key fobs to perform multifactor authentication.

Second, you can enable single sign-on across servers, which ensures that the user logs in once (for example at the organization's portal) and does not need to login again and again when accessing different servers that are part of the single sign-on domain.

Certificate-based authentication

In addition to what we previously described, you can supplement these authentication mechanisms by encrypting the communication channel between the user's Web browser and Domino using SSL. Either users use simple name and password authentication or session authentication, but through HTTPS instead of plain HTTP.

By doing so, during the authentication phase, it means that Domino Web Access users have the extra benefit of having the identify of the server authenticated by their certificate being presented and trusted (if it was created by a certificate authority that is recognized as a trusted root present in the Web browser's certificate store). It is not necessary for the client to have an x.509 Internet certificate if the client is set up for server-only authentication.

Server-only authentication is done through the use of the Secure Sockets Layer (SSL), which is set up on a protocol-by-protocol basis. This means that it is possible to enable SSL on either all protocols or only on specific protocols. For Domino Web Access, this entails enabling SSL for HTTP, but leaving other protocols (such as SMTP, POP3, and IMAP)

disabled. It is also necessary to enable the port for anonymous access; otherwise, Domino requires an Internet certificate or a name and password from the client.

At the high cost of managing all the user's x.509v3 certificates in the IT infrastructure, you can also use client certificate-based authentication. We explain this in Appendix C, “Domino as a certificate authority” on page 161. To enable client certificate-based authentication, in the Server document, go to the Ports tab → Internet Ports subtab → Web subtab. Set the Client certificate field to **Yes**, as shown in Figure 7-10.

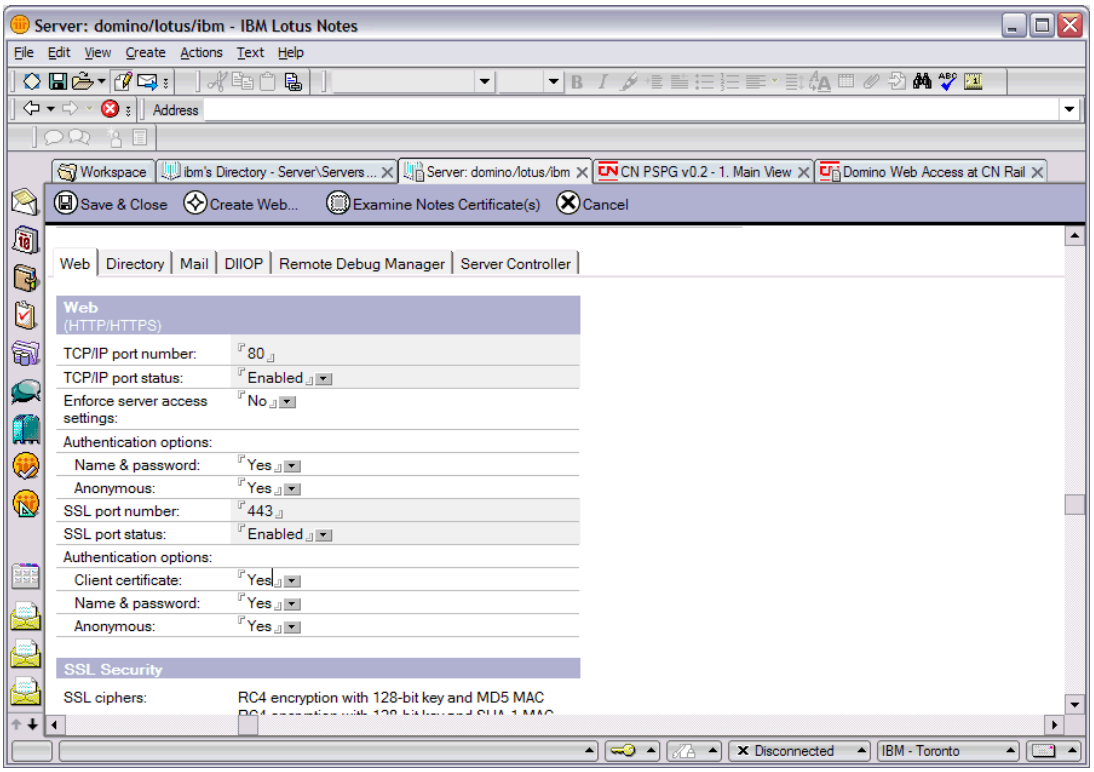


Figure 7-10 Enabling client certificate-based authentication

No matter the authentication setting (whether simple name and password authentication or session authentication), the user will not be prompted for a name and password, but for an x.509v3 certificate, as shown in Figure 7-11.

Figure 7-11 The client request form

Note: It is not necessary for the client to have an x.509 Internet certificate if the client is set up for server-only authentication.

For a Domino Web Access user to authenticate securely through HTTP to a Domino server using SSL, that user needs to use a supported Web browser that supports SSL and to have a trusted root certificate from a Domino or third-party certificate authority (CA).

To obtain a trusted root certificate for the Web browser from a Domino CA, the Internet client user performs the following steps:

1. Browse to the Domino Certificate Requests (for Domino 7) or Certificate Authority (Domino 5) application.
2. Select **Accept This Authority In Your Browser**.

If the trusted root certificate is for a third-party CA, follow the third-party CA's established procedure to properly merge the trusted root certificate for the CA. If both the client and server have certificates issued from the CA or already have a CA in common, this step is not necessary.

Because we are dealing with server-only SSL authentication here, the user has the alternative of skipping the CA database step and can simply use the SSL-secured server itself. There, the user can decide to trust that untrusted server when prompted. This is akin to the Notes cross-certification prompt when using a foreign Domino server. (Naturally, it is better to get the root certificate because then there is no need to do this on a server-by-server basis.)

We provide complete information about setting up a CA, creating a trusted root certificate, enabling SSL, and creating client certificates and picking them up for use with the Web browser in Appendix C, “Domino as a certificate authority” on page 161.

7.4 Browser Cache Management

Now that we have explained what happens when a user log *in* to Domino Web Access, let us see what specifically happens when a user logs *out* of Domino Web Access, specifically in regards to what can be done to ensure that this logout of Domino Web Access does not create a security vulnerability.

In the previous book, *Lotus Security Handbook*, SG24-7017, we explained that the logout function in Domino Web Access sends a logout command to the server to expire a session if session authentication is used. If basic authentication is used, the logon credentials are flushed from the Web browser. Domino Web Access also closes the browser window to prevent another user from clicking the back button to see the previous page, which might contain personal, sensitive information. In addition, Domino Web Access does some sophisticated things with caching algorithms to prevent the storage of information in the local browser cache in a format that someone could easily view.

However, the one thing we noted was that the secure function did *not* clear the browser's local cache. The only solution to ensure that the information contained in the cache was not accessible to unauthorized third parties was to either have the users clear the cache themselves (which is something most users do not do even when frequently reminded) or configure the browser to prevent the local storage of information (that is, temporary Internet files) after the session has terminated (which is something that can rarely be done in a kiosk of Internet café environment).

This issue is a thing of the past. With Domino Web Access in Domino 7, when the user logs out, in addition to Domino Web Access closing the browser and removing the user's logon credentials, it now clears private data from the browser's cache. This is even better than it seems at first glance: In Domino 7, with Browser Cache Management installed, even if the user does not explicitly log out, scrubbing will be done when the last Web browser window is closed. By deleting this data, Domino Web Access prevents an unauthorized user from using cached information to access the user's mail file.

In Microsoft Internet Explorer, you can use Browser Cache Management to improve the client-side performance and security of Domino Web Access sessions by controlling which entries are stored in the cache and which are removed when the Domino Web Access session ends. The removal of private data from the browser's cache and more secure data clearing capabilities are available only if the user accepts the Domino Web Access control.

Not all machines are created equal and not every location is guaranteed to have performance end-user machines or plenty of speedy bandwidth. Therefore, while it may laudable for organizations or businesses to restrict files that are left in the browser's cache for security reasons, there can be the problem that loading the Domino Web Access design elements for each session might adversely impact performance and cause needless delays before the user can make full use of the functionalities and features of Domino Web Access. To resolve this, you can, for example, leave the Domino Web Access design elements in the cache for performance reasons, but remove everything retrieved from mail databases for security reasons. Domino administrators can set the cache scrubbing level to remove all cache entries or only those related to the user's mail database.

Setting up Browser Cache Management

Browser Cache Management can be set in the Domino Web Access server's Configuration Settings document. Figure 7-12 on page 112 shows the settings on the Domino Web Access tab.

Note the Browser Cache Management section, which contain a number of parameters. We describe these options after Figure 7-12 on page 112.

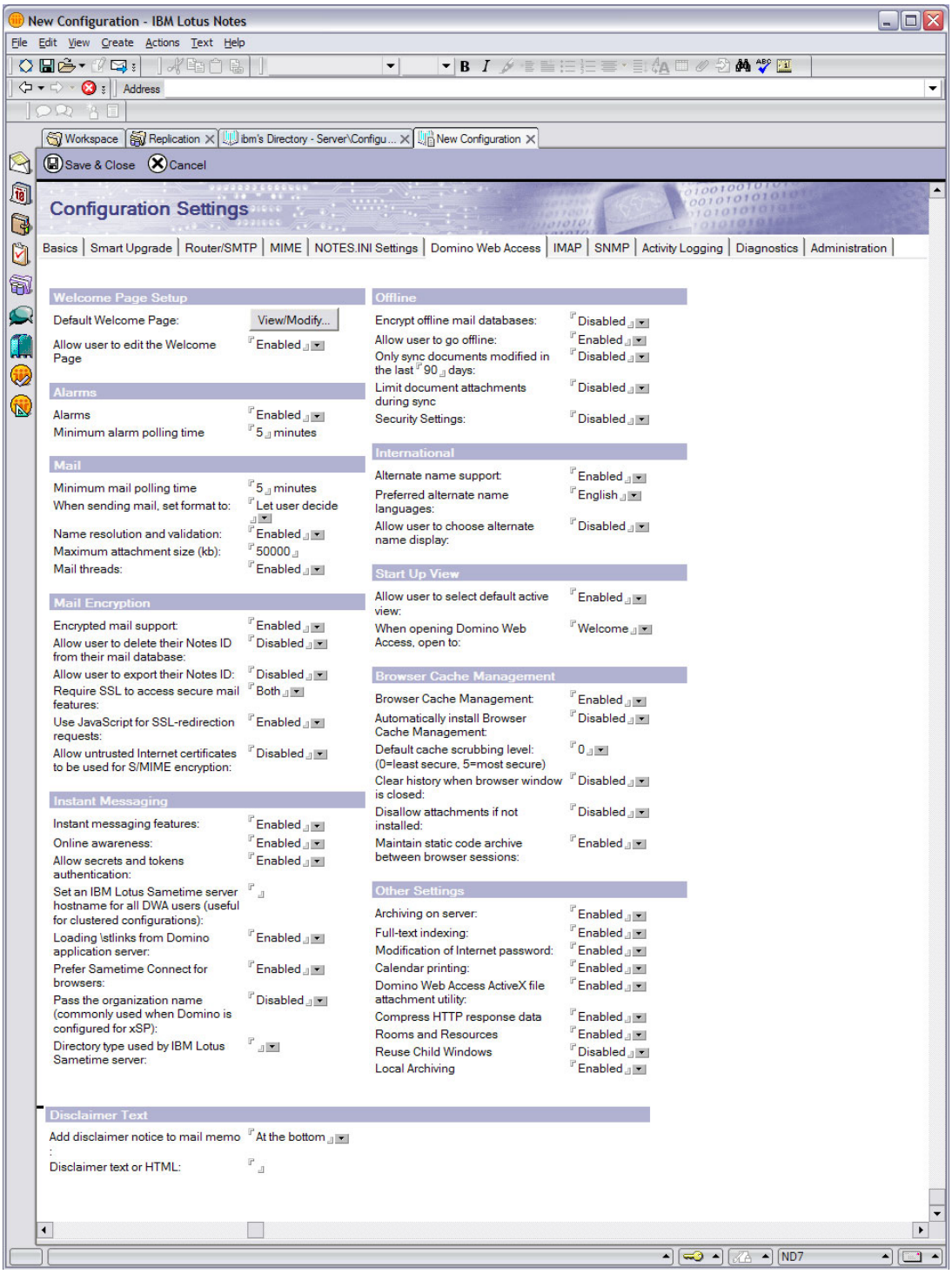


Figure 7-12 Configuration Settings for Domino Web Access

The Browser Cache Management section contains the following options:

- Browser Cache Management: Enabled | Disabled

  You can enable or disable Browser Cache Management for all Domino Web Access clients. This is not a per-user granular option, but applies as a whole for the entire user population.

- Automatically install Browser Cache Management: Enabled | Disabled

  You can enable or disable the ability to automatically install Browser Cache Management on all Domino Web Access clients. Again, this is not a per-user granular option, but applies as a whole for the entire user population.

- Default cache scrubbing level: 0 | 1 | 2 | 3 | 4 | 5 (0=least secure, 5=most secure)

  This field permits you to set the automatic cache clearing level for the Domino Web Access server. Table 7-1 explains the various scrubbing levels.

Table 7-1 Scrubbing levels

Level	Description
0	This is the default level value and is best for subsequent Domino Web Access performance. At this level, the Browser Cache Manager deletes all URLs that begin with the mail file path, except those that have a strategically placed KeepInCache (&KIC) argument. This argument marks page pieces that contain mostly design. Keeping these pieces in the cache offers a significant performance improvement when next using Domino Web Access. Examples of files deleted from the cache include parts to a MIME message retrieved through a separate URL or attachments opened when not using the Domino Web Access control.
1	At this level, the Browser Cache Manager deletes all URLs that begin with the mail file path. This is the best balance between Domino Web Access performance and security. It does not impact caching used by other Domino or other Web applications, nor does it impact caching of pages on the same Domino server or on other servers. Examples of files deleted from the cache (in addition to those listed for type 0) include: ► Most list and calendar view HTML top-level pages. ► The s_SessionInfo JavaScript™ page, which contains data about various preferences and relevant Domino Web Access configuration settings. This includes various variants of the current user's name (common name, abbreviated canonical name, full canonical name. ► The h_TOC JavaScript page, which contains information about the functional areas available for current user and initial URL information. ► The s_Outline, which contains information about folder names.
2	At this level, the Browser Cache Manager deletes all URLs in the cache that originate from the server host name, except for URLs that contain /iNotes/Forms7.nsf, the current forms file (or /iNotes/Forms6.nsf). This level offers the best balance of performance and security when the user might access other pages in Domino databases on the same server, or might access Domino Web Access and other reverse proxied intranet sites that might be cached (for example, linking to sites through QuickLinks in the Welcome page or through document links in received mail). For pages accessed through reverse proxy, the server refers to the reverse proxy server. This does not impact the performance of other Web sites the user visits after logging out. Examples of files deleted from the cache (in addition to those listed for types 0 and 1 include: ► Pages generated from any other Notes or non-Notes Web application on the server ► In a reverse proxy scenario, pages generated from any other Notes or non-Notes Web application on the same server or any other server that is reachable from a reverse proxy server ► Domino view icons
3	At this level, the Browser Cache Manager deletes all URLs in the cache that originate from the server host name. This provides more security, but impacts Domino Web Access performance negatively for subsequent logons because all cached static script and image pieces are deleted. It does not impact Web applications or pages generated from other servers, so does not negatively impact performance of other Web sites the user visits after logging out. Examples of files deleted from the cache (in addition to those listed for types 0-2) are URLs to /iNotes/Forms6.nsf, and the Domino Web Access static code pages, images, and style sheet.

Level	Description
4	At this level (which is a secure option), the Browser Cache Manager deletes all URLs in the cache except for URLs that contain /iNotes/Forms7.nsf, the current forms file (or /iNotes/Forms6.nsf). This provides the best balance of performance and security for Domino Web Access, but might negatively impact the performance of other Web applications or pages the user might be using. Examples of files deleted from the cache (beyond those listed for type 0-3) are any external Web pages loaded by the Domino Web Access Welcome page or traversed to through Domino Web Access or any other browser instance.
5	At this level (which is the most secure option), the Browser Cache Manager deletes all URLs in the cache. This provides the highest security, but has the greatest impact on Domino Web Access performance for subsequent logons because all cached static script and image pieces are deleted. Examples of files deleted from the cache (beyond those listed for all other types) are URLs to /iNotes/Forms7.nsf, the current forms file (or /iNotes/Forms6.nsf), and the Domino Web Access static code pages, images, and style sheet.

- Clear history when browser window is closed: Enabled | Disabled

  You can enable or disable the clearing of the browser history when the window is closed. This prevents access to previously displayed pages by unauthorized users.

- Disallow attachments if not installed: Enabled | Disabled

  You can enable or disable disallowment of attachments if Browser Cache Management has not been installed.

- Maintain static code archive between browser sessions: Enabled | Disabled

  You can enable or disable the ability to move static Domino Web Access design entries from the cache to a local folder. Design entries are restored to the Web browser cache when the browser is started again.

As mentioned previously, after the Browser Cache Management feature has been enabled, it is possible for the Domino administrator to choose whether to install it on Domino Web Access clients automatically or to give users the option of installing it.

If it is installed automatically, the first time a user accesses Domino Web Access, a Browser Cache Management system confirmation opens, prompting the user to close all browser windows for Browser Cache Management to take effect, as shown in Figure 7-13.

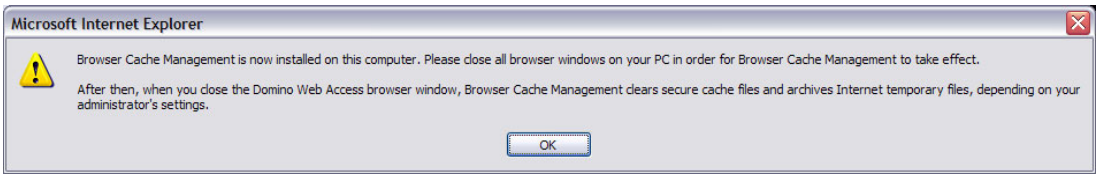


Figure 7-13 The Browser Cache Management confirmation message box

If Browser Cache Management is enabled, but it is not installed automatically, users can install (and uninstall) it using a Domino Web Access preferences (Preferences → Logout), as shown in Figure 7-14 on page 115.

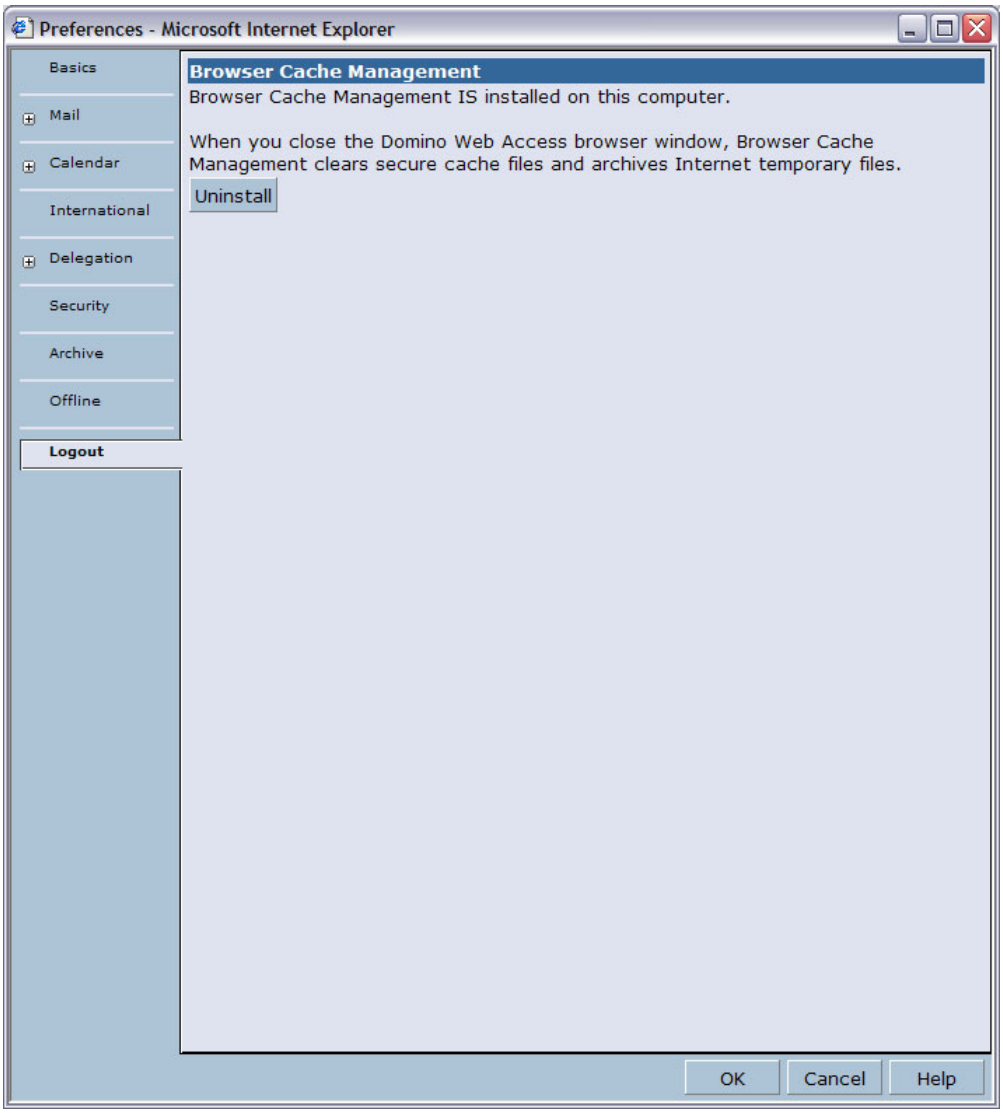


Figure 7-14 The Logout Preferences for uninstalling Browser Cache Management

If the uninstall button is pressed, Browser Cache Management will be uninstalled and the uninstallation confirmation message box opens, as shown in Figure 7-15.

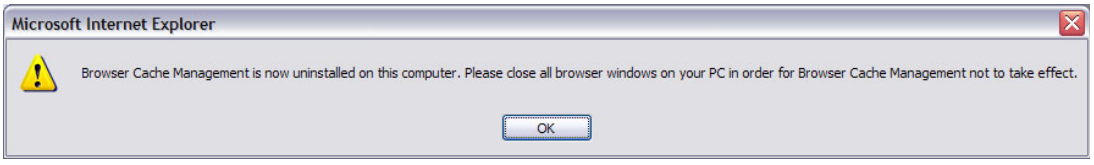


Figure 7-15 Confirmation of Browser Cache Management uninstallation

If Browser Cache Management is not enabled, this preference is not visible, as shown in Figure 7-16 on page 116.

Figure 7-16 Preferences: Basics

As an additional security measure, it is possible for the Domino administrator to prevent users who have not installed Browser Cache Management from adding or accessing e-mail attachments.

After the Browser Cache Management feature has been installed on a user's system, the cache cleanup occurs based on the cache scrubbing level set in the server's Configuration Settings document, which we described earlier. The user cannot change this.

The other parameters in the Domino Web Access Configuration Settings section control what is to be left and what is to be cleared.

The "Clear history when browser window is closed" configuration setting enables the option to clear the Web browser's history when the window is closed and also prevents access to previously displayed pages by unauthorized users. By default, this configuration setting is disabled.

The "Disallow attachments if not installed" configuration setting enables the option to prevent users from adding or accessing attachments in e-mail if Browser Cache Management is not

installed. Using this setting prevents users who have not installed Browser Cache Management from accessing or copying sensitive information in an attachment at an unsecured workstation. By default, this configuration setting is disabled.

Finally, the “Maintain static code archive between sessions” configuration setting enables the option to move static Domino Web Access design entries from the cache to a local folder on the machine so that they can be restored to the browser cache when the browser is started again. By default, this parameter is enabled.

Now that we have ensured proper login and have ensured that after we logout we do not leave anything behind, let us look at the secure messaging that occurs between the login and logout actions.

7.5 Secure messaging with Domino Web Access

Domino Web Access started offering secure messaging functionality in Release 6.5. In Release 7.0, it further extends this functionality. Let us review what was introduced in Release 6.5 and see how this is being built upon in Release 7.0.

7.5.1 Encrypted mail support in Domino Web Access 6.5

Domino Web Access offered secure mail support (or functionality) in Release 6.5. By storing their Notes ID within their mail file, users were now able to send and read encrypted mail messages.

This was done, server-side, by having the administrator edit the Configuration Settings document and select **Enable** in the Encrypted mail support field on the Domino Web Access tab.

Client-side, in order for Domino Web Access users to be able to encrypt or sign their mail messages, they had to make some changes within their user preferences, specifically to ensure that their mail file contained a copy of their Notes ID. If the mail file did *not* contain a copy of the user’s Notes ID, users had to import it into the mail through the Import Notes ID present on the Security tab in the user preferences dialog box. Then, users had to go to the Mail tab in the Preferences window and select the options that toggle signed and encrypted mail (a sure hint that a step had been skipped, namely the import of the Notes ID, is if buttons were unavailable). After completing these steps and saving the preferences, the Domino Web Access user would be able to sign and encrypt mail messages. Note that this was only needed if the user wanted signing or encrypting to be the default for each new message. Alternatively, users could select either option right at the top (below the toolbar) of each mail message.

A number of caveats existed with the use of encrypted mail in Domino Web Access. For example, the Domino Web Access user was limited to either decrypting an encrypted Notes mail message, or sending an encrypted Notes mail message to other users. There was no support for S/MIME encrypted or digitally signed mail messages.

7.5.2 New secure messaging features in Domino Web Access 7.0

In Release 7.0, Domino Web Access continues to support the secure messaging features introduced in Release 6.5 and now completes the set of secure messaging functionality by supporting S/MIME, greatly extending the security features of Domino Web Access for messaging.

It is now possible for users to make use of all the functionality offered by S/MIME to verify an S/MIME digital signature on a received message. If these users have an X.509 certificate in the mail file-based Notes ID, they can then also affix an S/MIME digital signature on messages they have composed in addition to decrypt received S/MIME messages. Outgoing messages can be S/MIME encrypted for recipients who have an X.509 certificate in the Domino Directory, in Domino Web Access contacts, or in any directory accessible through directory assistance.

It is worth noting that for an X.509 certificate to be used by Domino Web Access, an Internet cross-certificate must be issued from the user's organizational certifier to the certificate authority that issued the X.509 certificate. This Internet cross-certificate must be present in the Domino Directory. In addition, Domino Web Access also allows the use of untrusted certificates for encryption. The administrator much enable this in the Server Configuration document, as shown in Figure 7-17.

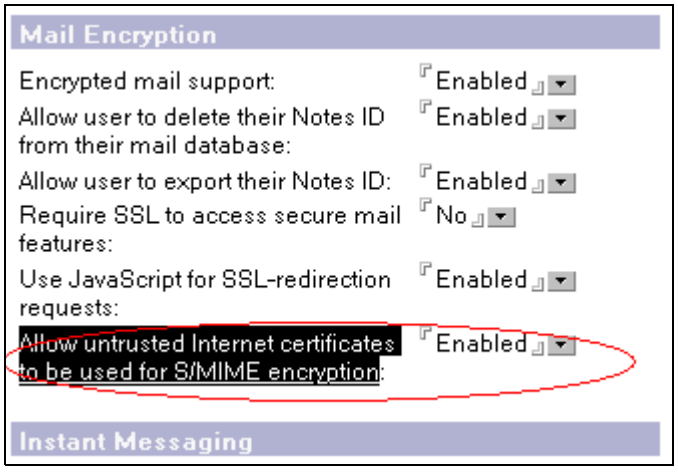


Figure 7-17 Allow untrusted Internet certificates setting

This done, users can choose to trust all Internet certificates (or alternatively, to be prompted) on each send that involves an Internet certificate that has not had a cross-certificate issued for it, as shown in Figure 7-18 on page 119.

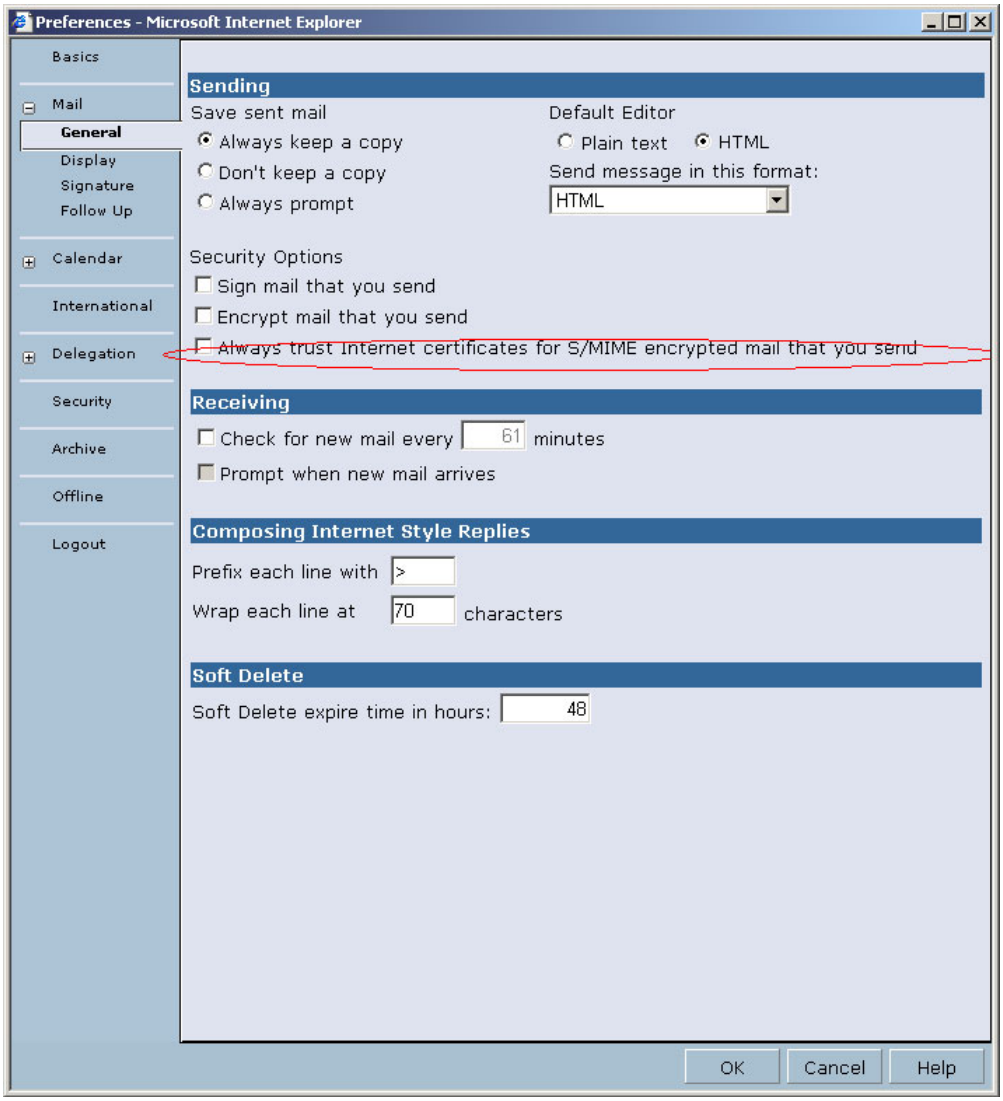


Figure 7-18 Domino Web Access user option to always trust Internet certificates for sent S/MIME mail

You can supplement secure messaging with the use of SSL. If an SSL connection is required for either the client or both the client and server, Domino Web Access users cannot read or send encrypted messages when connected through HTTP. If a user is connected through HTTP, the user must switch to HTTPS when accessing the encrypted message on the server. This switch occurs automatically when sending encrypted mail. The user will be prompted to switch when reading encrypted mail.

7.5.3 Domino Web Access secure messaging with S/MIME

To enable users to send encrypted or digitally signed messages, perform the following steps:

1. Enable relevant Domino Web Access Configuration Settings fields.
2. Add an Internet certificate and cross-certificate for encrypted S/MIME messages.
3. After completing the previous two steps, the next step, naturally, is to exchange S/MIME e-mails.

The rest of this section provides a guide for each step.

Step 1: Enable relevant Domino Web Access Configuration Settings fields

To allow Domino Web Access users to encrypt and digitally sign e-mail messages, first the Domino Administrator enabled both the “Encrypted mail support” and the “Name Resolution and Validation” fields on the Domino Web Access tab of the server's Configuration Settings document. This is no longer necessary with Domino Web Access 7.0. If a user attempts to send an encrypted message, the required name lookups will be done regardless of the configuration setting.

We showed in Figure 7-12 on page 112 the whole set of configuration settings available to configure Domino Web Access. Of specific interest to us is the Mail and Mail Encryption sections of the configuration settings for Domino Web Access.

The relevant field of the Mail section is:

- Name resolution and validation: Enabled | Disabled

  You can enable or disable alternate name lookups, similar to “type-ahead” in Notes. It lets users resolve ambiguous names and use alternate names by checking names against a contact list or Domino Directory.

  Note: It is mandatory that this field be enabled for the Domino Web Access secure mail feature, except in Domino Web Access 7.0, where, as stated previously, the required name lookups will be done regardless of the configuration setting.

The relevant field of the Mail Encryption section is:

- Encrypted mail support: Enable | Disable

  You can enable or disable the ability to allow users to use a stored Notes ID to read encrypted mail. The user's ID must be stored in the mail database. The default value is Enabled.

Other fields of interest in the Mail Encryption section include:

- Allow user to delete their Notes ID from their mail database: Enable | Disable

  You can enable or disable the ability to allow users to delete their Notes ID from their mail database. The default value is Disabled.

- Allow user to export their Notes ID: Enable | Disable

  You can enable or disable the ability to allow users to export and save their ID in a separate file. The default value is Disabled.

- Require SSL when reading encrypted mail: No | Client | Both

  This field offers three options to define the SSL requirement:

  – No: To treat encrypted mail the same as unencrypted mail
  – Client: (default) To require the browser client to use SSL, but not the server
  – Both: To require both the browser client and the server to use SSL

  The default setting is Both.

- Use JavaScript for SSL-redirection requests: Enable | Disable

  You can enable or disable the use of JavaScript to redirect SSL.

  Note: Some reverse-proxy servers do not properly fix 302 redirects. If so, enabling this option might help. Do not enable this option unless necessary.

- Allow untrusted Internet certificates to be used for S/MIME encryption: Enable | Disable

  You can enable or disable the ability to allow users to use an untrusted Internet certificate for S/MIME encryption. The default value is Disabled.

Step 2: Add an Internet certificate and cross-certificate

Next, to allow Domino Web Access users to encrypt and digitally sign e-mail messages, ensure that the sender has the recipient's Internet certificate in the Domino Web Access Contacts, Domino Directory, or LDAP directory. The sender must also have a cross-certificate issued for the recipient or for the certifier who issued the recipient's Internet certificate, except when the proper configuration setting has been enabled and where the relevant preference setting is in place.

If a cross-certificate is issued to the recipient's CA, it is possible for the sender to send encrypted messages to all recipients who have certificates issued by that CA if the sender has the recipients' Internet certificates. If the Internet certificate is stored in a Domino Directory in another domain or in an LDAP directory, the directory needs to be accessible using directory assistance.

To add an Internet certificate and cross-certificate, the recipient must send an S/MIME digitially signed e-mail to the user. The user can then reply to the message, having access to the recipient’s public key. However, this information needs some refinement. If a Domino Web Access user receives an S/MIME signed message and adds the sender to Contacts, the sender's x.509 certificate is added to the Contact record. What is really important to note is that, unless the “add sender to contacts” step is done, the certificate will not be available for subsequent sends.

Step 3: Exchange S/MIME e-mails

The following example shows the exchange of S/MIME e-mails between a Domino Web Access user and a Lotus Notes user.

In Figure 7-19, the Lotus Notes user has sent a digitally signed S/MIME e-mail to the Domino Web Access user. The Domino Web Access user now has the user's public key that can be used to send an encrypted e-mail. The indication that this is a signed message is visible by the presence of a little ribbon to the right of the sender's name.

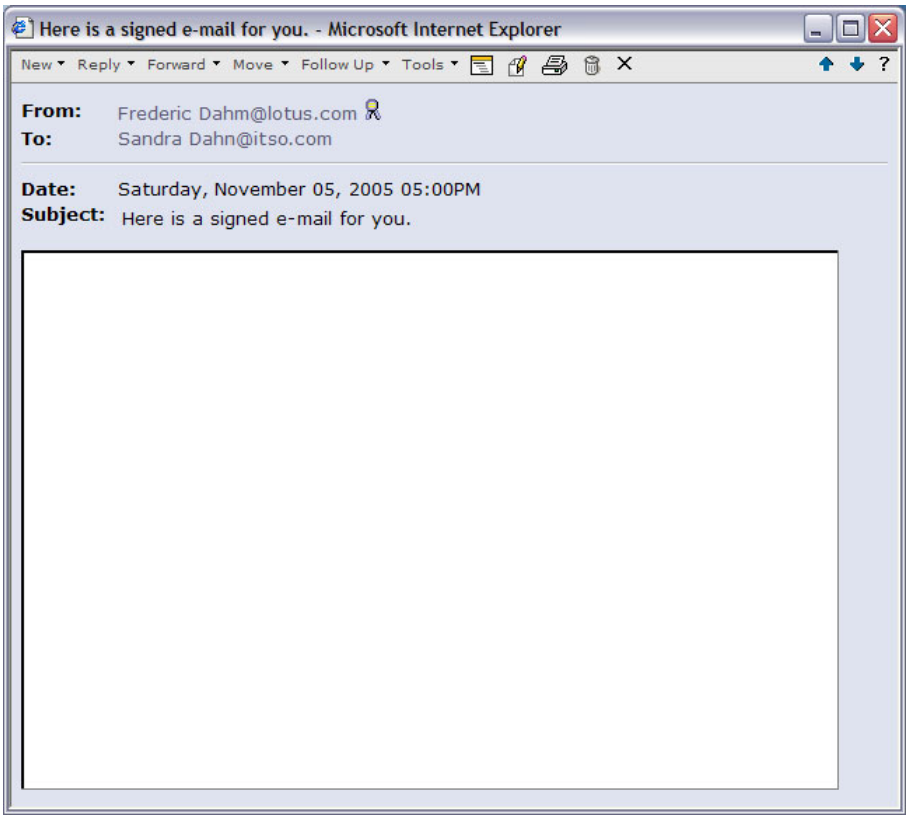


Figure 7-19 Receiving a signed S/MIME message

However, for the time being, the Domino Web Access user decides to address a digitally signed S/MIME message to the Lotus Notes user, as shown in Figure 7-20.

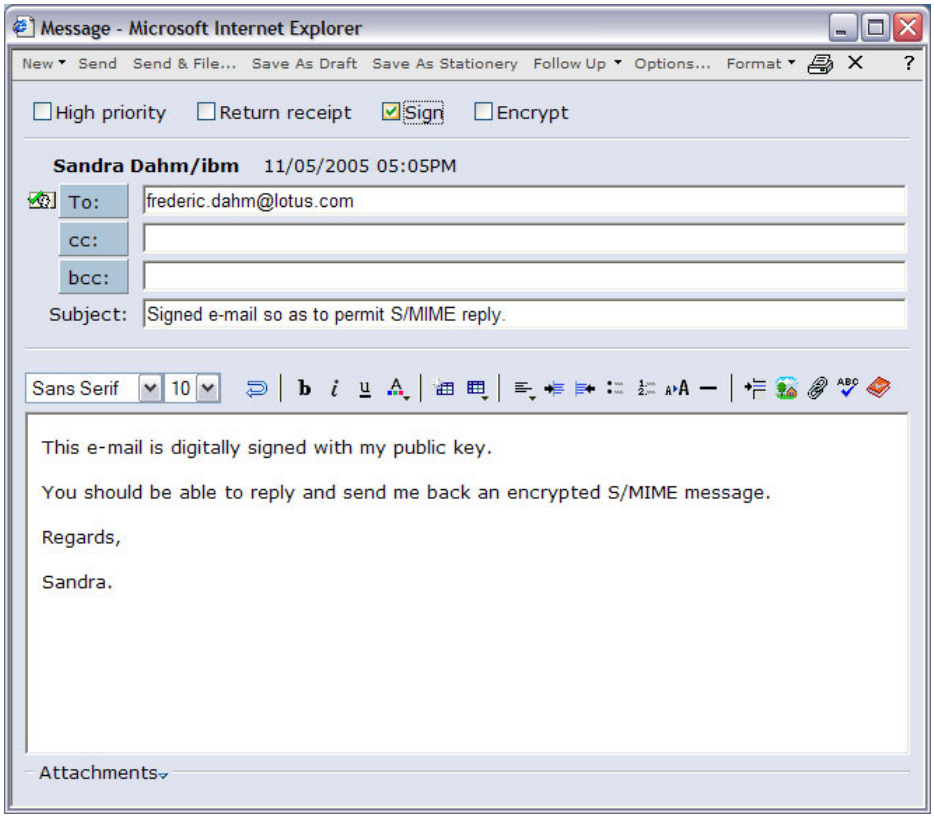


Figure 7-20 Replying with a digitally signed S/MIME message

The Lotus Notes user receives the digitally signed S/MIME message, as shown in Figure 7-21.

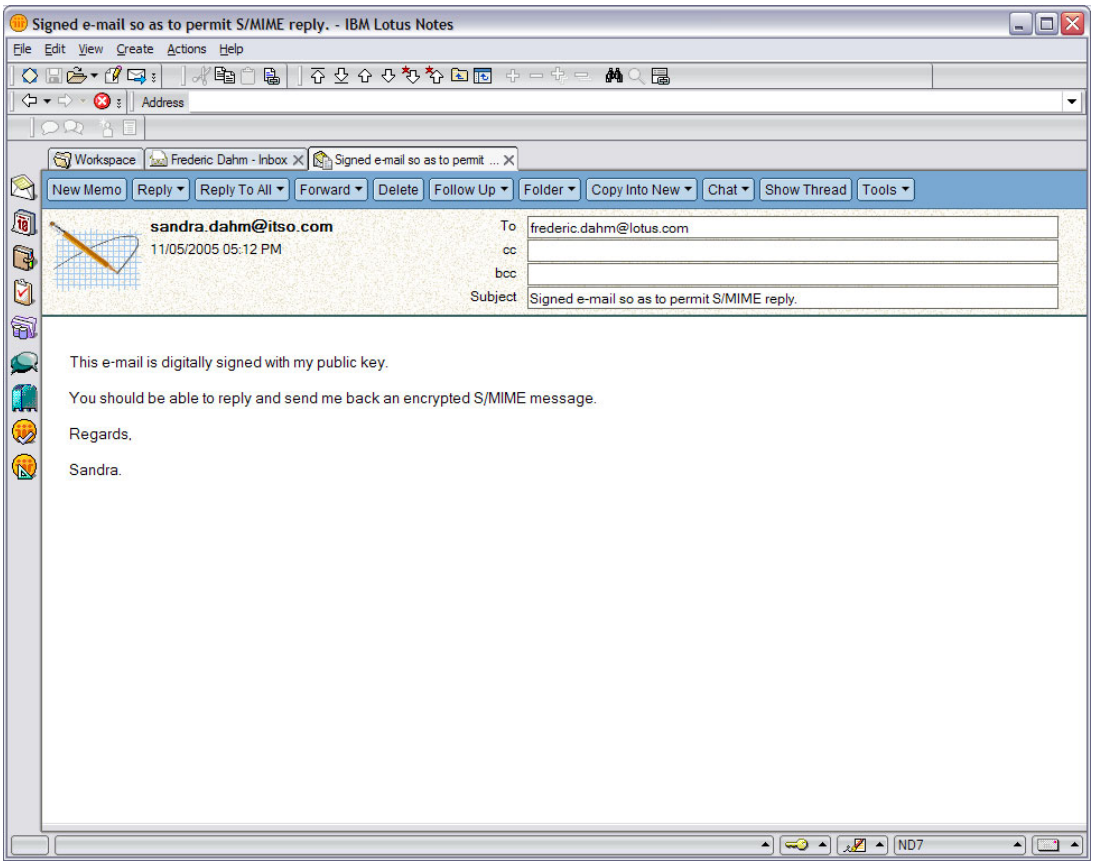


Figure 7-21 Receiving a digitally signed S/MIME message

The Lotus Notes user decides to address an encrypted S/MIME message to the Domino Web Access user, as shown in Figure 7-22.

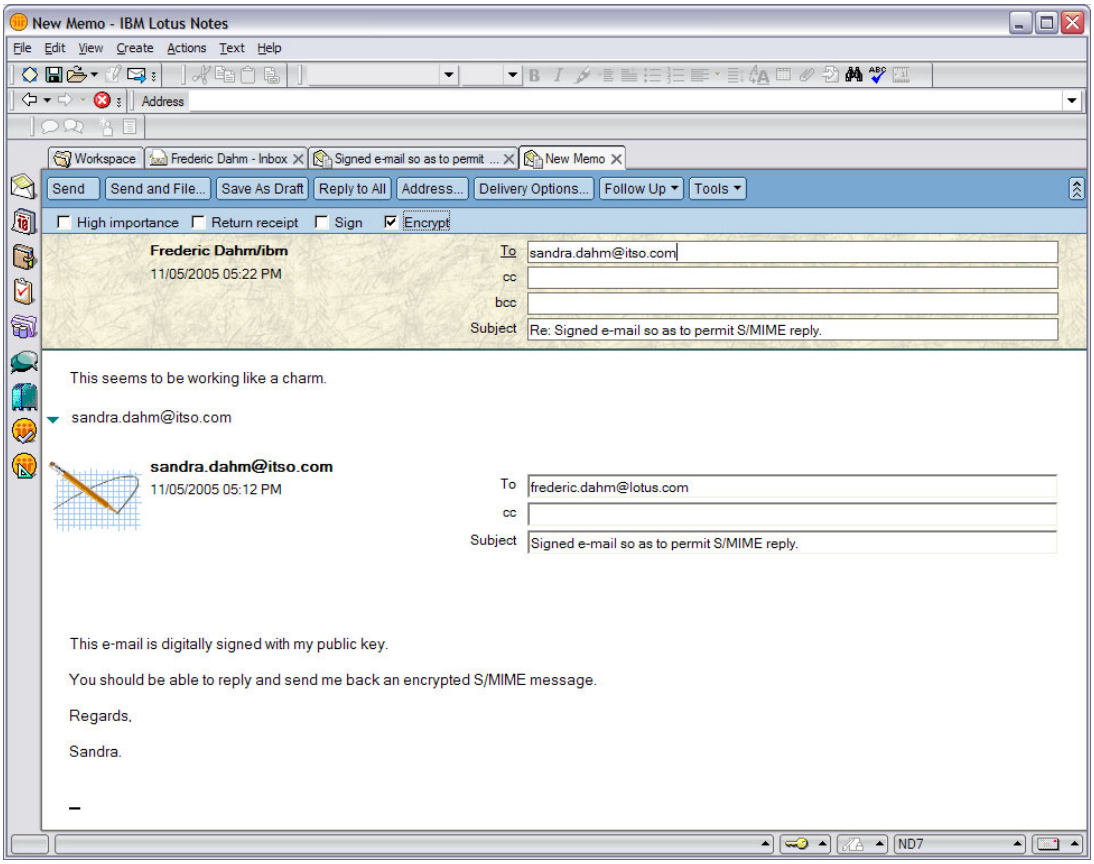


Figure 7-22 Replying with an encrypted S/MIME message

The Domino Web Access user then receives an encrypted S/MIME message from the Lotus Notes user, as shown in Figure 7-23. The indication that this is an encrypted message is visible by the presence of a little lock to the right of the sender's name.

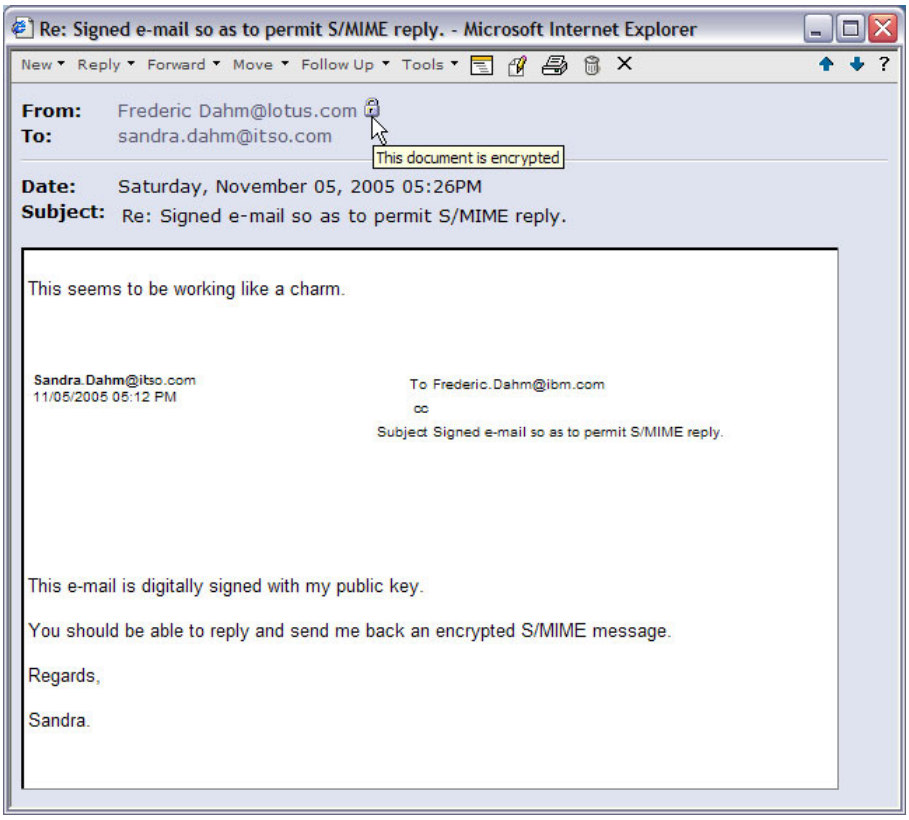


Figure 7-23 Receiving an encrypted S/MIME message

At this stage, the Domino Web Access user can send an encrypted S/MIME message. However, we do not show this here.

Note that when both Notes and S/MIME signing and encryption are possible, Domino Web Access uses S/MIME signing and encryption by default. This might cause problems in a mixed environment that includes both Domino 7 and pre-Domino 7 servers. Pre-Domino 7 servers do not support S/MIME, so messages sent S/MIME signed and encrypted cannot be verified or decrypted.

We recommend that you use the NOTES.INI setting `iNotes_wa_SecMailPreferNotes=1` when Domino Web Access 7.0 users are exchanging encrypted mail with Domino Web Access 6.x users and these users have been issued x.509 certificates. This setting is not supported offline.

7.5.4 Additional Domino Web Access security considerations

There are a number of client-side security factors with Domino Web Access similar to those of a standard Notes client. In addition to our previous logout discussion, the physical security of your machine is critical (do not leave your browser session logged in while unattended, lock it down with a Kensington lock or similar device, and so on). Encrypt local (offline) databases, and establish Domino offline security policy documents.

Finally, we must discuss the issue surrounding security liabilities that come from using a browser itself as a client. By using browser client technologies as the main method of communication with the server, there is the increased potential for users to be subject to malicious code in the form of JavaScript, Java agents, ActiveX® controls, and the like. To prevent bad agents and code from being triggered by the clients, Domino Web Access by default has an Active Content Filter in place that parses the HTML content of every mail message and rewrites it prior to having it display in the browser. This can affect server performance, so we have a NOTES.INI flag that enables you to disable it if wanted.

To disable the Active Content Filter, set the following property in the NOTES.INI file and restart HTTP:

```
iNotes_WA_DisableActCntSecurity=1
```

Setting this parameter to `0`, or commenting out or deleting the line in the NOTES.INI file will re-enable the filter.

There are also some deployment differences between Notes and Domino Web Access that need to be considered:

- Recovery authority: Domino Web Access does not support recovery authority (that is, ID recovery) unless it is already in the ID mailed to the user.
- Imported Notes IDs: Notes IDs cannot be smartcard enabled.
- Certificates: Domino Web Access looks for certificates first in the Domino Directory and then in the Contacts.
- Cross certificates: Domino Web Access looks for cross-certificates only in the Domino Directory. If Domino Web Access is used, any required cross-certificates must be created in the Domino Directory.
- Multiple domains: If multiple domains are being administered, use directory assistance for an Extended Directory Catalog on the server. Do not use a Condensed Directory Catalog (CDC) on the server.
- Offline: If a directory catalog is being used, it must be enable for encrypted mail.

This completes our review of the new security features of Domino Web Access. The next chapter covers a topic of interest to both Lotus Notes clients and Domino Web Access users: the new functionality that helps win the fight against spam, or unsolicited e-mail.

8

Spam control using Domino 7

This chapter addresses new and enhanced features in Domino 7 that help you to control the increasing the spam mail problem.

For more detailed documentation about spam and Domino, refer to *Lotus Domino 6 spam Survival Guide for IBM @server*, SG24-6930, available at:

http://www.redbooks.ibm.com/abstracts/sg246930.html

When you need to create a spam filter solution for your organization, you can use the Domino functionality to manage your e-mail. We provide a summary of all related configuration options in this chapter plus an in depth look at the new whitelist features. In some cases, you might prefer third-party help. This can be in the form of hardware, software, or a service. We do not include any considerations for third-party tools or services in this book.

With the introduction of Domain Name System (DNS) whitelists in Domino 7, we include information about how to set up a DNS that can be used for whitelist DNS lookup. The topic of DNS deserves its own book, but for the purpose of this book, we made it work with only a few steps and (almost) no prior DNS knowledge. We use the Microsoft DNS Server and the DNS Manager software to set up the required files. It is important to understand that this setup does not consider any security implications that are required when hosting a DNS. The whitelist DNS here is intended for low to medium mail traffic in a completely protected intranet. Keep in mind that only your SMTP server needs to access the DNS whitelist.

The new whitelist options and the blacklist enhancement are now accessible in the server rules and mail file rules.

Spam filters are not just a technical problem, you need policies that deal with all your e-mail needs. For more details about this topic, see 8.3.1, “E-mail policies and user education” on page 133.

8.1 SMTP

Understanding how to control spam starts with understanding Simple Mail Transfer Protocol (SMTP) and its inherent weaknesses. Example 8-1 shows how you can interact with an SMTP server using a Telnet client and send a short but technically complete and legal e-mail message to that server.

Lines starting with > are commands from the sending SMTP server with the name dieter.stdi.com.

Lines starting with < are replies from the receiving SMTP server with the name whistler.groofty.com.

Example 8-1 Simple SMTP communication

```
>Telnet whistler.groofty.com 25
<220 groofty.com ESMTP Service (Lotus Domino Release 7.0) ready at Thu, 3 Nov 20 05

>HELO dieter.stdi.com
<250 groofty.com Hello dieter.stdi.com ([9.33.85.84]), pleased to meet you

>MAIL FROM: <dstalder@stdi.com>
<250 dstalder@stdi.com... Sender OK

>RCPT TO: <system@whistler.groofty.com>
<250 system@whistler.groofty.com... Recipient OK

>DATA
<354 Start mail input; end with <CRLF>.<CRLF>
>From: dstalder@stdi.com
>Subject: Test SMTP Message
>Test line 1
>Test line 2
>.

<250 Message accepted for delivery
<221 groofty.com SMTP Service closing transmission channel
```

From Example 8-1:

- When the connection is established, the Domino server writes connection information into the server log file and replies with the 220 message.
- HELO command: The sending server sends a greeting with its own host name. Spammers frequently will forge a fake name, The receiving server replies with a 250 message.
- MAIL FROM command: The sending server sends the sender's e-mail address. Spammers almost always forge addresses. The Domino server writes this information into the server log file (if logging level=verbose) and replies with a 250 message.
- RCPT TO command: The sending server sends the recipient (or recipients) e-mail address. The Domino server writes this information into the server log file (if logging level=verbose) and replies with a 250 message. The message will be delivered to any legal recipient addresses that were sent with this command.
- DATA command/block: The sending server sends the message headers and body. This is the actual e-mail message and can have several different formats. It is important to understand that the receiving server does not deliver to the addresses listed in the To, Cc, and Bcc headers.

- At the end of the transmission, the DATA block is accepted with a 250 message and the transmission is closed. The Domino server writes the disconnect message into the server log file. At this point, the message is in the server's MAIL.BOX file, awaiting delivery.

8.2 Spammer techniques

Spammers use different tricks to force their e-mail into your inbox. They want you to open the e-mail and click the link. Their goal is to sell you a product or service, to trick you and try to get your private information, or sometimes to deliver a virus or trojan horse program that they can use to take over your computer to send more spam. No matter how reputable the sender appears, the burden is on you to differentiate between a good and a bad message.

This section list spammer techniques and is by no means complete. To stay current on spammer techniques, a good place to start is The Spamhaus Project, at:

http://www.spamhaus.org

E-mail harvesting

Harvesting refers to the collection of e-mail addresses from Web sites, Usenet sites, LDAP directories, or any other place where an e-mail address is stored.

Remove instructions that should take your e-mail address off the distribution list are sometimes just a tool to validate your e-mail address. Even opening an e-mail can be enough to validate your e-mail. Images in the e-mail are retrieved from the spammers Web sites and can include codes to identify the recipient.

E-mail validation

E-mail validation is another form of harvesting that uses tools to verify e-mail address by connecting to the SMTP server. These tools do not send mail messages. The tools send the RCPT TO command and check the reply. Depending on the reply, the e-mail address is accepted by the SMTP or rejected. Example 8-2 shows the three basic replies, the 250 Sender OK, the 550 No such user, and the 554 Relay rejected.

Example 8-2 SMTP communication to test RCPT TO

```
>Telnet whistler.groofty.com 25
<220 groofty.com ESMTP Service (Lotus Domino Release 7.0) ready at Thu, 3 Nov 20 05
>HELO spammer.stdi.com
<250 groofty.com Hello spammer.stdi.com ([9.33.85.84]), pleased to meet you

>MAIL FROM: <spammer@groofty.com>
<250 spammer@groofty.com... Sender OK

>RCPT TO: <albert@toronto.stdi.com>
<250 albert@toronto.stdi.com... Recipient OK

>RCPT TO: <bob@toronto.stdi.com>
<550 bob@toronto.stdi.com... No such user

>RCPT TO: <bob@ibm.com>
<554 Relay rejected for policy reasons.

>QUIT
<221 groofty.com SMTP Service closing transmission channel
```

No messages are sent; the communication terminates before the DATA block is sent. The server log file shows a connection established and a connection disconnected (with 0 messages). Without the Logging level set to Verbose, there is no indication anywhere that an e-mail verification took place. The Verbose logging prints the MAIL FROM and RCPT TO commands in the server log file. See 8.5.10, “Logging level” on page 143.

Directory attacks and name guessing

Domino by default will deliver a messages based on the first name or the last name as long as they are unique. Spammers often send e-mail to the most popular names and hope for a match. Finding a list of the top 500 first names or last names is only a click away. Try to google baby names or census information and you get plenty of information.

Denial of Service, or DoS attacks, overload your SMTP server or network with excessive traffic.

This type of attack can hit anyone, well-known organizations and government agencies. For example, you installed a server and you did not close the relay on the SMTP server. The operating system software might not be updated with the latest fix. In some cases, it only takes a few minutes until an attacker finds your exposed system and gains control over it. In just a few minutes, your system could be loaded up with thousands of spam messages in the out queue. Under this kind of load, your system will not be able to handle your regular mail service.

Open relay

Relaying is a normal SMTP feature that allows a server to accept a message and then pass it on to another server. A large percentage of spam mail is distributed using relay servers that do not check whether the sender is actually authorized to use them. If your SMTP server is an “open relay” that accepts mail and redistributes the messages without authentication, spammers *can* abuse it for their purpose.

Spammer friendly ISPs

Internet service providers (ISPs) provide the connectivity between continents, between cities, and eventually to your office and home. Most ISPs in North America and Europe have an acceptable use policy (AUP) in place that makes the distribution of large amounts of e-mail a violation that results in the termination of your service, but there are exceptions. Most spam originates in the United States, but the Internet enables spammers to pick and choose ISPs to work with anywhere in the world. It is quite common for U.S.-based spammers use servers operated by ISPs in other countries where the AUP is not as strict.

Forged RECEIVED header

During the routing of a message, the receiving SMTP adds a RECEIVED header to the message. Domino stores these headers in fields named RECEIVED in the e-mail message. An Internet message has at least one RECEIVED header (added by your own SMTP server) and sometimes many more. These headers are routinely forged by spammers and should rarely be trusted as evidence of the actual origin of spam messages.

Example 8-3 on page 133 shows three headers. The first header (or last SMTP in the routing path) is your own SMTP and the information can be trusted. In this example, the connecting server’s IP address is 32.97.182.142 with the name e2.ny.us.ibm.com. This is also the IP address used for *inbound connection control* and *DNS whitelist and blacklist* checking.

Example 8-3 RECEIVED header

```
"from e2.ny.us.ibm.com ([32.97.182.142]) by notes5.stdi.com with ESMTP id
2005092615444927-4653 ; Mon, 26 Sep 2005 15:44:49 -0400"

"from d01relay04.pok.ibm.com (d01relay04.pok.ibm.com [9.56.227.236]) by e2.ny.us.ibm.com
(8.12.11/8.12.11) with ESMTP id j8QJkTDg009592 for <dstalder@stdi.com>; Mon, 26 Sep 2005
15:46:29 -0400"

"from d01av02.pok.ibm.com (d01av02.pok.ibm.com [9.56.224.216]) by d01relay04.pok.ibm.com
(8.12.10/NCO/VERS6.7) with ESMTP id j8QJkTEZ063346 for <dstalder@stdi.com>; Mon, 26 Sep
2005 15:46:29 -0400"
```

Note: The IP address shown in the first RECEIVED header is the IP address that is used for inbound connection control and DNS whitelist and blacklist checking.

Fake sender addresses and domains

The sender e-mail address can be anything. The SMTP standard requires very little beyond validation of the technical format of messages, so there is no technical barrier to forgery. Spammers abuse this to pose as a trusted sender, for example, ibm.com. When you open the message, you might find that the message did not originate from IBM. Two initiatives will solve this problem. One is the DomainKeys initiative that uses a form of digital signature in the e-mail message. The other is the Sender Policy Framework (SPF) that uses DNS to identify the sending host. For more details about these proposed standards, see 8.7, “The future of spam” on page 154.

Phishing and pharming

Phishing gained popularity in early 2004. Phishers try to convince you that they need to verify some personal information (credit card, bank PIN number, social security number, and so on). The link they provide in the e-mail will direct you to a decoy Web site that is a look-a-like of the official Web site of an organization they hope you trust. All information entered by the unassuming user will be used for criminal purposes. See The Anti-Phishing Working Group (APWG) for more information, available at:

http://www.antiphishing.org

8.3 Avoiding spam

The best solution in the battle against spam is to avoid it in the first place. If only your closest business contacts knew your e-mail address, all would be fine. You would know exactly who sends e-mail and can adjust the configuration to only allow messages from these domains. This model does not work anymore. There are ways, however, to protect your e-mail addresses.

8.3.1 E-mail policies and user education

The best way to avoid spam is to educate users and implement policies. Several Web sites will help you define your policy. The following Web site is a good place to start:

http://www.emailreplies.com

The key issues for any organization are:

- E-mail format

  Define the structure of the e-mail message: the greetings, body format, and signature. Define the distribution list: CopyTo and BCC.

- Use of e-mail system

  Define the private use of e-mail system. Define how to communicate with customers and suppliers. Define the turnaround when replying to e-mail.

- How to deal with spam

  Tell users how to respond to spam mail. Describe how to report illegal and offending e-mail. Several countries have systems in place to report child pornography and messages with criminal intent. Contact your local authorities for details.

Users are often unclear when it comes to how to deal with spam. Sometimes spam is funny and users send a copy to their collegues and friends. Sometimes spam is upsetting and users forget about the rules and click the remove-me instructions. And sometimes spam has a criminal intent. Make sure that your users clearly understand how to react to these messages.

Tip: Always provide a positive action for users when they deal with spam. Users might get frustrated about the spam message and need something more then just the delete button. Provide a reporting application where they can express their anger. In many cases, that is all they need and are not really interested in any follow up. If they do not have a place to express their frustration, they will often find one.

8.3.2 Prevent e-mail harvesting

If your e-mail address is on your Web site, you are getting spam.

A bold statement? Unfortunately not. Web site harvesting is a very popular spammer technique. So, how do you know if you Web site was scanned? Review the Domino Web Server log, which contains all client requests and the requests from indexing services. They might be for a search engine or for a spammer who extracts only e-mail addresses and URLs.

Figure 8-1 shown an example is from the Domino Web log. The request came from 66.249.65.81, which is assigned to Google Inc. (as per WHOIS). The Browser Used field shows Googlebot/2.1 and both the IP address and the browser is a good indication that this page was retrieved for the Google search engine.

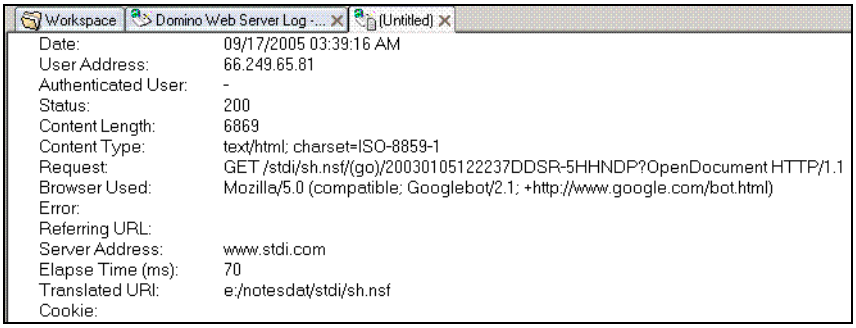


Figure 8-1 Domino Web Server Log

Having search engines scan your Web site is a good thing. If you do not want your site indexed, or portion of your site excluded from the index, create a ROBOTS.TXT file in your

root directory. Well-behaved scan engines will read the file and skip the pages defined in the file. Refer to the following Web site for more details:

http://www.robotstxt.org

Solutions to prevent harvesting

In this section, we describe how to prevent harvesting.

Contact Us application

Create a Contact Us application where visitors fill out a simple form and use this instead of posting e-mail addresses on your Web site. Instead of receiving an e-mail message, you have the comment in a database. You can distribute the information to any person or group in your organization and create a workflow to guarantee that all inquiries are replied to in timely fashion.

Important: In the long run, the Contact Us application is the only solution that will keep your e-mail addresses from being harvested.

JavaScript to write e-mail address

Another potentially less effective solution is to write the e-mail address with JavaScript. The e-mail address is not in its native format, but broken apart as JavaScript parameters. This can work reasonably well today. Keep in mind that these scripts can be resolved to an e-mail address by all browsers. If the harvesting engines today do not translate the JavaScript, they will be able in the very near future.

Example 8-4 shows a sample JavaScript string.

Example 8-4 JavaScript to hide e-mail address

```
In HTML HEAD
    <script language="JavaScript" type="text/javascript">
    var ma;
    function gem(mu,md){
    ma=mu+'@'+md;
    document.write('<a href="mailto:' + ma + '">' + ma + '</a>');
    }
    </script>

In BODY
    <script language="JavaScript">
    gem("dstalder","stdi.com");
    </script>
```

8.3.3 Open relay

Correctly securing your systems prevents spammers from using it to spam other people. Although it might not reduce the number of spam you receive in your organization, it does save your administrator from following up on mail sent to abuse in your organization.

Note: Abuse and postmaster are two required addresses. This is documented in Technote 1106677, “Email Messages Addressed to “postmaster” or “abuse” are Being Delivered to the Domino Server Administrator.” These messages are delivered to the administrator as configured in the Server document unless they are configured as a mail recipient. This Technote is available at:

http://www.ibm.com/support/docview.wss?uid=swg21106677

Spammers use agents to scan the Internet trying to find open relays. When they find your unsecured server, they will send spam through your server.

Relay controls

Figure 8-2 shows the following rely controls settings:

- Allow messages to be sent only to the following external internet domains

  List all Internet domains for which you receive mail. For example, enter @stdi.com if you only accept mail for stdi.com. Enter stdi.com if you accept mail for all stdi.com and subdomains (for example, @mail.stdi.com, @ca.stdi.com).

- Deny messages to be sent to the following external internet domains

  By default, Domino adds an asterisk (*) in the field. This prevents Domino from relaying messages to any external Internet domain.

Inbound Relay Controls		Inbound Relay Enforcement	
Allow messages to be sent only to the following external internet domains:	stdi.com @kai-shin.com	Perform Anti-Relay enforcement for these connecting hosts:	External hosts
Deny messages to be sent to the following external internet domains: (* means all)	*	Exclude these connecting hosts from anti-relay checks:	
Allow messages only from the following internet hosts to be sent to external internet domains:		Exceptions for authenticated users:	Allow all authenticated users to relay
Deny messages from the following internet hosts to be sent to external internet domains:(* means all)	*		

Figure 8-2 Relay controls

8.4 Detecting spam

In this section, we describe how to detect spam.

8.4.1 Directory attacks

Attacks typically result in lots of mail that is not delivered, and this mail gets stuck in the server mailbox. The Domino configuration determines how a directory attack affects your system:

- Invisible, but not unaffected. If you configured the Domino server to only accept mail where there is a user in the Domino Directory (8.5.4, “Inbound intended recipients controls” on page 140), you might not know that you are being attacked. The spammer might still occupy listener threads and make your system unavailable for legitimate e-mail.
- Lots of held messages. If you configured the Domino server to hold undeliverable mail, these messages end up as held (8.5.9, “Hold undeliverable messages” on page 143).

- Lots of dead messages. If you have the default settings applied, messages that are not delivered will be returned (non-delivery report). A large number of spam messages uses invalid sending addresses. When the Domino server replies with the non-delivery notification, the sender does not exist. With no place to go, the message status is set to dead. The problem is not necessarily the dead message in the mailbox. Before it changed the status, the Domino server tried to send the non-delivery notification. This uses system resources and can keep the server unavailable for your important e-mail traffic.

When you review the server log file, you might find a message that has multiple recipients with only one or two valid recipients. When you look at the held messages in the server mailbox, you might notice that several of these messages have the same or similar subject lines.

Example 8-5 Server log file: List of recipients

```
10/23/2005 09:58:01 AM  SMTP Server: mx04.ca.mci.com (142.77.2.24) connected
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <carpenter@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <chandler@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <chapman@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <cohen@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <conner@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <cruz@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <cummings@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <daniel@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <daniels@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <dawson@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <delgado@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <dennis@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <douglas@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <duncan@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <dunn@stdi.com>
10/23/2005 09:58:01 AM  SMTP Server: Recipient: <ferguson@stdi.com>
```

Most directory attacks are slowly growing over time, from a few messages a day to a few messages per minute. But like all spam problems, everybody has a variation of the different problems.

8.4.2 Phishing and pharming

Phishing and pharming refers to e-mail where the sender's goal is to steal personal information from the recipient. In most cases, the sender poses as an official institution that needs to verify your personal information. The best protection against these types of scams is user education.

Find more details at:

http://www.antiphishing.org

8.5 Blocking spam

In this section, we describe techniques to block spam.

8.5.1 Whitelist and blacklist options

Domino R7 introduces the whitelist option, an important feature when it comes to controlling spam mail. An effective whitelist enables you to use a stricter blacklist without an excessive risk of falsely identifying messages as spam.

Note: The blacklist and whitelist validation happens after the inbound relay enforcement. Messages have to be destined for your domain before they are validated against the lists.

Both blacklist and whitelist options have the private and DNS option. The private option enables you to enter a domain name or an IP address.

The DNS blacklist feature is not new, but private blacklist and both DNS and private whitelist are new in Release 7.

Figure 8-3 shows the DNS blacklist, DNS whitelist, private blacklist, and private whitelist settings.

DNS Blacklist Filters		DNS Whitelist Filters	
DNS Blacklist filters:	Enabled	DNS Whitelist Filters:	Enabled
DNS Blacklist sites:	sbl.spamhaus.org	DNS Whitelist Sites:	query.bondedsender.org
Desired action when a connecting host is found in a DNS Blacklist:	Log and tag message	Desired action when a connecting host is found in a DNS whitelist:	Log and tag message
Custom SMTP error response for rejected messages:			
Private Blacklist Filter		**Private Whitelist Filter**	
Private Blacklist Filter:	Enabled	Private Whitelist Filter:	Enabled
Blacklist the following hosts:	groofty.com [192.168.1.*]	Whitelist the following hosts:	stdi.com [207.176.156.*]
Desired action when a connecting host is found in the private blacklist:	Log and tag message	Desired action when a connecting host is found in the private whitelist:	Log and tag message
Custom SMTP error response for rejected messages:			

Figure 8-3 DNS Blacklist, DNS Whitelist, Private Blacklist, and Private Whitelist

In the private blacklist and whitelist, as well as all other fields in the Server document that allow entry of host names or IP addresses, individual IP addresses must be surrounded by square brackets, for example, [192.168.100.128]. You can also specify a range of addresses such as [192.168.100.128-255], and you can use wildcards such as [192.168.128-144.*].

A new feature in Domino 7 enables you to enter addresses in Classless Inter-Domain Routing (CIDR) format such as [192.168.0/16], but you will need to take care with this format because the Domino interpretation of CIDR format is slightly different from the standard. The CIDR string [207/8] should be equivalent to [207.0/8], but Domino does not interpret [207/8] correctly, so when your objective is to be listing an entire class A, B, or C address range, be sure to include the ".0" in the CIDR strings.

These fields can become very large and entering addresses into them directly can result in situations where the data in the Server document exceeds the summary buffer limit. You can avoid this by entering group names into these fields and entering host names and addresses (in the same formats) into the groups.

The four options are executed in the following sequence:

1. Private whitelist
2. Private blacklist
3. DNS whitelist

4. DNS blacklist

The private and DNS blacklist actions have the following options: Log only, Log and tag message, Log and reject message.

The private and DNS whitelist actions have the following options: Silently skip blacklist filters, Log only, Log and tag message.

The tag option adds a field to the message. The blacklist adds $DNSBLSite, and the whitelist adds $DNSWLSite. When the DNS whitelist or blacklist tags the message, the field stores the URL. When the private whitelist or blacklist tags the message, the field stores either PrivateWhitelist or PrivateBlacklist. You can use mail rules to detect these fields and take action.

The log option adds an entry in the Mail Routing Events on the server log file. Example 8-6 shows the four possible options.

Example 8-6 Whitelist and blacklist log entries

```
SMTP Server: Remote host 192.168.1.221 (list.groofty.com) found in blacklist at
PrivateBlacklist
SMTP Server: Remote host 192.168.1.221 (groofty.com) found in blacklist at sbl.spamhaus.org
SMTP Server: Remote host stdi.com (192.168.1.221) found in whitelist at PrivateWhitelist
SMTP Server: Remote host list.stdi.com (192.168.1.221) found in whitelist at
query.bondedsender.org
```

8.5.2 Inbound connection controls

All inbound connection controls are performed based on the connecting IP address. None of the SMTP fields are used in the validation. This validation happens before the server receives the MAIL FROM command. Figure 8-4 shows the inbound connection controls settings.

Note: Virus filters, mail gateways, or third-party services might sit between the Domino SMTP server and the Internet. In these cases, the connecting IP address is always the same.

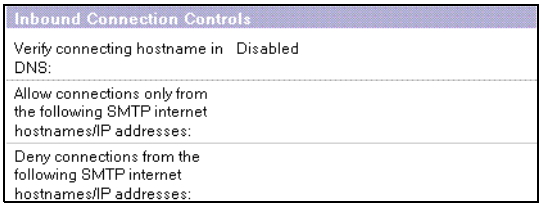


Figure 8-4 Inbound Connection Controls

Note the following options:

- Verify connecting hostname in DNS field

  Domino performs a reverse lookup to the DNS. The domain must have a PTR record in the DNS. The PTR record maps an IP address to a host name.

  Note: PTR records are not mandatory for correct Internet mail routing. Many organizations do not maintain these records in the DNS. Use caution when enabling this feature.

- Allow connections only from the following SMTP internet hostnames/IP addresses and Deny connections from the following SMTP internet hostnames/IP addresses

  This specifies host names and IP addresses that are allowed to connect or not allowed to connect. When you enter a host name, Domino performs a reverse lookup to find the name associated to the IP address.

  Note: This feature is similar to the private whitelist, but not as flexible because it does not provide options tagging or long-only options.

8.5.3 Inbound sender controls

All inbound sender controls are performed based on the SMTP header field MAIL FROM.

Note: Not all header fields are stored in the mail document. The MAIL FROM field in the SMTP header is logged in the server log file and the message tracking log. The address from the MAIL FROM field can be different from the sender in the user's mail file.

Figure 8-5 shows the following inbound sender controls options:

- Verify sender's domain in DNS

  Domino verifies that the domain of the sender exists. The domain is taken from the message header MAIL FROM. A valid MX, CNAME, or A record must exist in the DNS.

- Allow/Deny messages only from the following external addresses/domains

  Enter the addresses and domains. Domino compares the MAIL FROM in the message header with addresses and domain entered here.

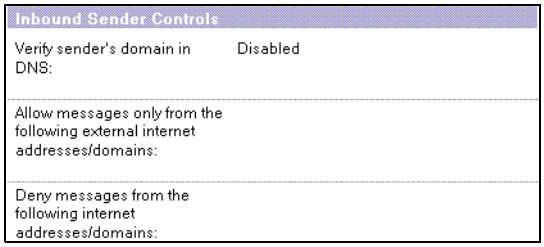


Figure 8-5 Inbound Sender Controls

8.5.4 Inbound intended recipients controls

All inbound intended recipients controls are performed based on the SMTP header field RCPT TO.

Note: Not all header fields are stored in the mail document. The RCPT TO field in the SMTP header is logged in the server log file and the message tracking log. The address in the RCPT TO field can be different from the SendTo (or CC) in the user's mail file.

Figure 8-6 on page 141 shows the following inbound intended recipients controls options:

- Verify that local domain recipients exist in the Domino Directory

  Domino verifies that the recipient exists. The recipients e-mail is taken from the message header RCPT TO.

- Allow/Deny messages intended only for the following internet addresses

  Enter e-mail addresses of users. For example, you enter dstalder@stdi.com in the Deny field, only mail that matches this address format is denied; this user still receives mail under dieter.stalder@stdi.com.

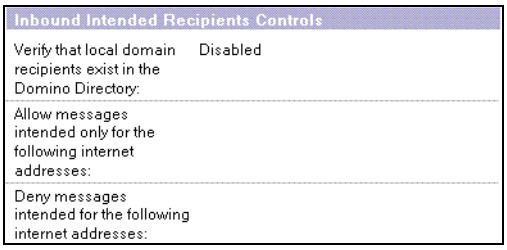


Figure 8-6 Inbound Intended Recipients Controls

8.5.5 Server rules

Server rules now include the BlackList tag and WhiteList tag under the Conditions and Stop Processing under the Actions. The blacklist reads the $DNSBLSite field, and the whitelist reads the $DNSWLSite field. The Domino server stores the URL of the DNS blacklist or whitelist site, or the string PrivateBlacklist or PrivateWhitelist.

In Figure 8-7, the administrator uses the new Stop Processing rule when a message gets tagged by either a blacklist or whitelist. The message can be processed by the user mail rules, leaving the ultimate decision to the user.

The server rules option has the following conditions and actions:

- Conditions: sender, subject, body, importance, delivery priority, To, CC, BCC, To or CC, body or subject, internet domain, size (in bytes), all documents, any attachment name, number of attachments, form, recipient count, any recipient, blacklist tag, whitelist tag
- Actions: journal this message, move to database, don't accept message, don't deliver message, change routing state, stop processing

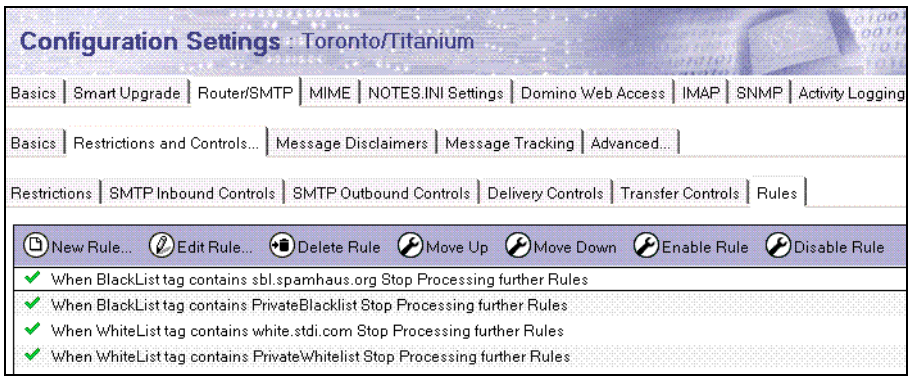


Figure 8-7 Server Rules

8.5.6 Mail file rules

Mail file rules now include the BlackList tag and WhiteList tag under the Conditions and Stop Processing under the Actions. The blacklist reads the $DNSBLSite field, and the whitelist reads the $DNSWLSite field. See Figure 8-8 on page 142.

The mail file rules option has the following conditions and actions:

- Conditions: sender, subject, body, importance, delivery priority, to, cc, bcc, to or cc, body or subject, internet domain, size (in bytes), form, blacklist tag, whitelist tag, all documents.
- Actions: move to folder, copy to folder, send copy to, set expire date, change importance to, stop processing, Delete (don't accept message).

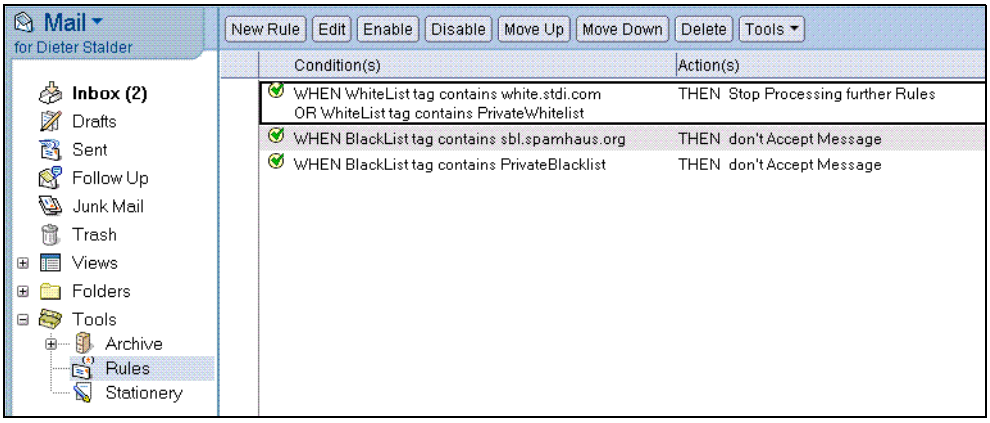


Figure 8-8 Mail file rules

8.5.7 Address lookup

The address lookup determines how the e-mail address is mapped to all the name variations within Domino.

When Domino resolves a recipients address, all the combinations from the $Users view in the Domino Directory are valid. You will receive e-mail by your first name only, last name only, and any combination that is listed in the User name field of the Person Document. To prevent spam mail that is sent to random first names or last names, select the **Fullname only** option in the address lookup, as shown in Figure 8-9.

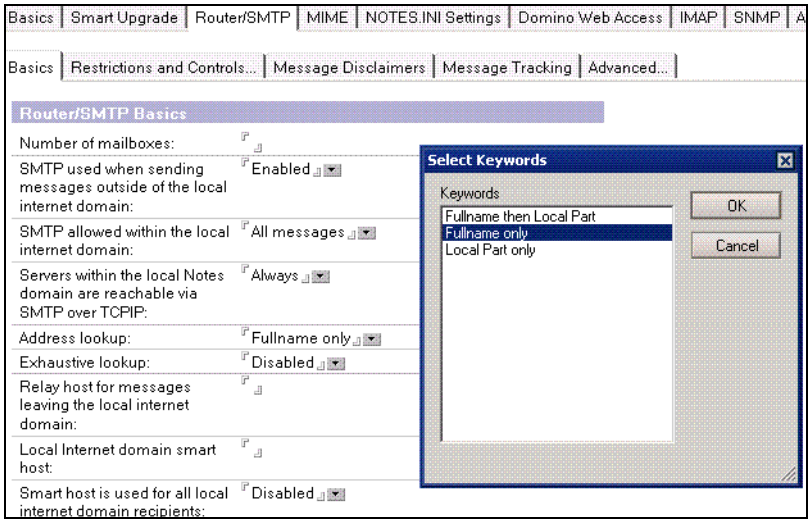


Figure 8-9 Address lookup options

You need to add all acceptable Internet e-mail address to the User name list. Figure 8-10 on page 143 shows an example.

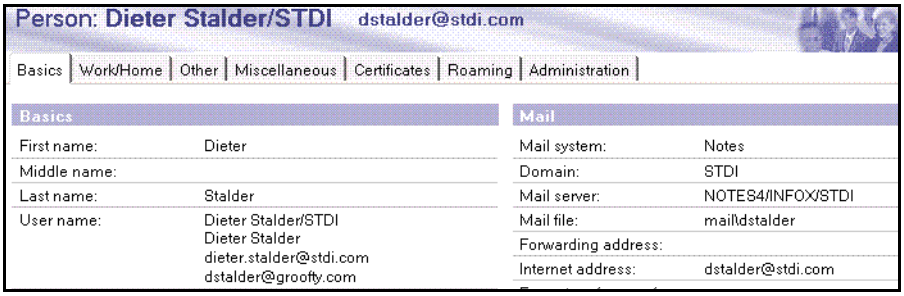


Figure 8-10 Person document

When an Internet e-mail address does not exactly match the options in the $Users view, the message is rejected with the “User not found in Domino Directory” message.

For more details, refer to “How to Stop Incoming Mail Addressed to Just the Last Name,” Technote 1090405, and “SMTP Mail Is Received by User in Spite of User's Different Internet Address in Person Document,” Technote 1192804.

8.5.8 Primary directory only

In most installations, Internet e-mail is only accepted for recipients in the primary directory. If you have directory assistance configured, the default Internet mail routing also includes the secondary directories.

Set the Restrict name lookups to primary directory only field to **Enable**, as shown in Figure 8-11.

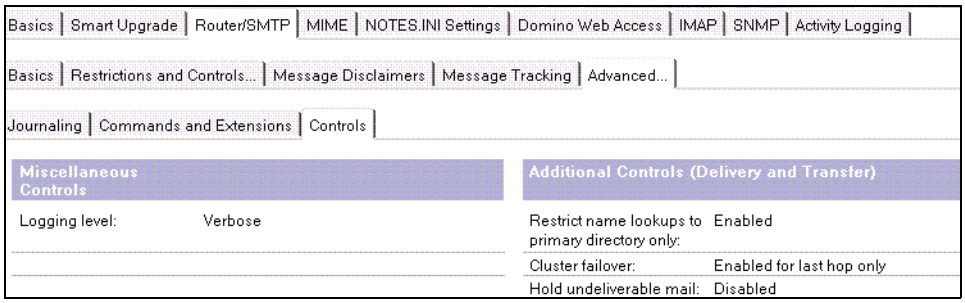


Figure 8-11 Primary directory only lookup

8.5.9 Hold undeliverable messages

All messages are held, even Notes messages. Domino does not differentiate between your users that should get a non-delivery report or a spam message to a non-existing user. The difference is that your user would like to get the non-delivery report so that they can respond to the error. Figure 8-11 shows this option.

8.5.10 Logging level

You might ask why the logging level is listed as part of the security consideration. The answer is simple: Domino will write the MAIL FROM and RCPT TO content into the server log file. In many cases, this is the only source where you can trace the routing of an Internet message. Remember, the address fields in the message header do not have to match the ones in the

message body. Selecting the verbose option will show you all address fields from the message header. Figure 8-11 on page 143 shows this option.

Note: When an SMTP server connects to Domino, things happen (or do not happen) in a specific order. This order is partly determined by the rules of the SMTP protocol and partly determined by the configuration settings that you have set up for your server. This section describes the sequence of events that can happen and the configuration settings that make them happen:

1. SMTP inbound control: Greeting
   a. Inbound connection detected.
      Logged in server log file.
   b. Reverse DNS lookup.
      When any of the inbound connection controls are enabled.
   c. Connection controls:
      i. Verify host exists in DNS.
         Inbound connection controls → verify connecting host name in DNS.
      ii. Allow/deny.
         Inbound connection controls → allow/deny connections only from the following SMTP Internet host names/IP addresses.
   d. Flag possible relays.
      Successful AUTH resets flag.
   e. Whitelist and blacklist filters:
      i. Private whitelist
      ii. Private blacklist
      iii. DNS whitelist
      iv. DNS blacklist
   f. Send greeting.
      Sends greetings.
2. SMTP inbound controls: MAIL
   a. Sender control:
      i. Verify sender's domain in DNS.
         Inbound sender control.
      ii. Allow/deny.
         Inbound sender control.
   b. Enforce:
      Sender controls.
      Connection controls.
      DNSBL filters: reject message.
3. SMTP inbound controls: RCPT
   a. Parse recipient address.
      RCPT TO in message header.
   b. Recipient controls:
      i. Allow/deny.
      ii. Verify local domain recipient.
   c. Enforce:
      Recipient controls.
      Relay controls.
4. SMTP inbound controls: DATA
   a. DNSBL filter.
   b. System mail rules.
   c. Deposit message into MAILBOX.

8.6 Review strategies using R7 features

Which configuration options are most effective for you? There is no clear answer. You have to know how mail arrives at your Domino server. If a gateway is between the Domino server and the Internet, you cannot use the connection controls or the DNS blacklist and whitelist options. The connecting IP address is always your gateway address.

Put all the configuration settings together in two scenarios, a reject all spam scenario and a accept all spam scenario. With the introduction of the whitelist option in R7, we review the DNS whitelist and how you can maintain your own DNS whitelist, or your own DNS blacklist for that matter.

8.6.1 Accept all spam or reject all spam?

There are good reasons why accepting all spam (or even better, all e-mail) is the best solution. One reason is a configuration where the Domino server is not directly receiving mail from the Internet. For example, you might have an SMTP backup server that is provided by your ISP, or an SMTP server run by a service provider or gateway server within your organization. In all these cases, rejecting a mail message does not really solve the spam problem, it just moves the responsibility to deal with the spam message to another position.

Another reason to accept all spam is that you will have an excellent cross reference to fine tune your spam filters. When Domino rejects a message, you often do not know the exact reason, particularly when it comes to mail rules. If you keep the message, you can always review the message and update your filter. This is especially true when it comes to false positives, the messages that are flagged as spam by mistake.

There are also good reasons to reject all spam. There will be few or no dead messages, and you will save bandwidth because the message body is never transmitted.

You can decide to accept or reject all messages in a few configuration settings. Choosing one method over the other does not mean that users get more or less spam messages. The difference is that the mailbox on the server and possibly quarantine files will accumulate large quantities of spam.

Consider the following checklist:

- Inbound intended recipients controls

  This setting is very effective to control the inflow of messages that are not correctly addressed. The Domino server terminates the connection before the message body is received, saving bandwidth and freeing up inbound threads. Keep in mind that enabling this setting has precedence over the whitelist options. In short, if it is not in the Domino Directory, it will be rejected. See 8.5.4, “Inbound intended recipients controls” on page 140. This setting does not reduce the spam mail to the individual user; it is an administrative option.

  Note: Enabling the inbound intended recipients controls validates an e-mail address based on the RCPT TO field. The SMTP server will reply with “250...Recipient OK” and “550...No such user” to the spammer harvesting tools.

- Hold undeliverable messages

  The default is disabled. If you do not want messages that are not delivered returned to the sender address, enable this option. Keep in mind that all messages are held, even the ones sent by your users. You need to check the mailbox file on the server frequently and either release the held messages or delete them. See 8.5.9, “Hold undeliverable messages” on page 143. This setting does not reduce the spam mail to the individual user; it is an administrative option.

- Address lookup

  The default is Fullname and Local Part. To avoid delivery of messages by only the first name or last name, change this option to **Fullname only**. If you accept more then one e-mail domain, update the Person documents with the different e-mail addresses. See 8.5.7, “Address lookup” on page 142. This setting can reduce the mail sent to users, because the exact address format is required.

- Whitelist and blacklist

  Choose a blacklist provider that meets your criteria. Start with the **Log and tag** option and switch to the **Log and reject message** option after you monitored the results. This setting can reduce the mail sent to users.

  Setup a whitelist with your customers domains. Educate your users to create mail file rules with the whitelist tag. This setting can improve the quality of the mail being delivered.

- Private whitelist/blacklist or connection control?

  You can enter the domain name (or IP address) in either of them. The difference is in the action. The whitelist/blacklist enables you to tag a message; the connection control only accepts or rejects. You can use the private blacklist with the log and tag option to test a setting. When you are satisfied with the result, you can copy the domain name (or IP address) to the Connection Control section.

- Server rules

  Review your existing rules and update actions with the new Stop Processing option. This option is useful to terminate the rule validation when you have a match. Any further rules validation is stopped, and this can improve server performance.

8.6.2 Set up your own whitelist DNS

DNS blacklists and DNS whitelists work identically from a technical point or view. The SMTP server sends a query and the DNS returns a Not-Found or an IP address. For DNS blacklists and whitelists, the IP returned is usually 127.0.0.1, but Domino does not differentiate between the IP addresses returned; any address is considered a match.

When you consider setting up your own whitelist DNS, you have to collect the IP addresses of your senders that you want to whitelist. This task is not as simple as looking up a domain and entering the corresponding IP addresses. There is no current, widely-deployed standard that requires an entry in any DNS for a sending SMTP server. Keep in mind that the sending SMTP server for an organization does not have to be the same as the receiving SMTP server, and frequently is not, so the registered Mail Exchanger (MX) records for a domain are not necessarily the ones you want to whitelist. The only source you have is the server log file where the IP address is logged at connection and disconnection time. See Example 8-7.

Example 8-7 SMTP connect and disconnect

```
SMTP Server: mail2.kai-shin.com (9.33.85.111) connected
...
SMTP Server: mail2.kai-shin.com (9.33.85.111) disconnected. 1 message[s] received
```

The next step is to add this IP address to the whitelist DNS.

How does the DNS lookup work?

DNS lookup for whitelists, blacklists, and any other query works basically the same. The DNS client (that is your SMTP server) calls the DNS server that is configured in your operating system. In Microsoft Windows, the DNS setup is part of the network configuration.

Consider the following points:

- Sending an e-mail: When the SMTP needs the IP address for a domain, it issues a standard DNS query for the domain asking for the MX entry (MX entries are the mail exchange entries that accept mail). The DNS responds with all the MX entries and their associated address records. The SMTP picks the entry with the highest priority and the mail transmission starts.
- Receiving an e-mail: By default, the SMTP server accepts e-mail for you domain without the need to query the DNS. Several configuration options will initiate a DNS lookup:
  - Entering a host name in the Relay and Connection Controls, the SMTP server does a reverse name lookup (IP address to host name translation).
  - Enabling DNS blacklist and whitelist options.
  - Entering a host name in private blacklist and private whitelist, the SMTP server does a reverse name lookup (IP address to host name translation).
  - Enabling the option “Verify connecting hostname in DNS,” the SMTP server does a reverse name lookup (IP address to host name translation).
  - Enabling the option “Verify sender’s domain in DNS,” the SMTP server does a name lookup (name to IP address translation).
- DNS whitelist/blacklist lookup: When the SMTP does the lookup, it checks if the sender is in any of the whitelists or blacklists. The server composes a query with the IP address. If you have a whitelist configured with the name white.stdi.com and the connecting servers IP address is 192.168.1.217, the SMTP server issues a forward lookup request (name to IP address). The request format is 217.2.168.192.white.stdi.com (note the reversed IP address). The DNS replies with “No such name” or the corresponding IP address. The IP address is usually 127.0.0.1 (loopback address).
- Private whitelist/blacklist lookup: When the SMTP needs to compare the sender with a host name entry in the private whitelist or blacklist, the SMTP needs to convert the senders IP address into a host name first. It issues a query to return the name for the IP address (PTR record or reverse lookup). The DNS responds with the host name, not the domain name. The host name is now compared to the private whitelist or blacklist and tagged if a match is found. Keep in mind that not all servers have a PTR entry in the DNS.

Set up a whitelist or blacklist DNS server

You can use any DNS server software to act as a whitelist or blacklist DNS. The whitelist and blacklist queries are standard DNS queries. Our example here shows the Microsoft DNS server. The Microsoft DNS management software is not optimized to handle whitelist entries, but the DNS server works well when the entries are added manually. Another DNS server option is BIND from Internet Systems Consortium. You will find DNS software for many Microsoft Windows platforms and UNIX® platforms. For more information, see:

```
http://www.isc.org
```

If you plan to run a whitelist DNS for high-volume queries, refer to the following Web sites:

```
http://www.corpit.ru/mjt/rbldnsd.html
http://cr.yp.to/djbdns.html
```

Both products are designed for large volume blacklists (and whitelists).

Important: If you are not familiar with the basic working of a DNS, coordinate your work with the network administrator. You need a correctly configured zone file.

In our example, we create a stand-alone DNS server that will act as only a whitelist DNS server (white.stdi.com). To work correctly, the parent domain (stdi.com) needs to point to this subdomain with a name server (NS) entry. On the Microsoft DNS server, this is the New Delegation option. For an alternate option to maintain only one DNS server, we can add these entries to the stdi.com zone file, or create a subdomain on the same server.

Microsoft DNS server setup and configuration

Our examples here use the Microsoft Windows 2000 Server. We configured the Windows 2000 Server as a stand-alone server and installed the DNS server software.

After installing the product, open the DNS Manager software in Control Panel. We use the DNS Manager software to configure the basic DNS options. After we have the required files, we will make further changes manually.

Perform the following steps:

1. Launch the DNS Manager.
2. The first task is to create a zone. A zone refers to domains or subdomains. In our case, we use white.stdi.com domain as our whitelist DNS. See Figure 8-12.[1]

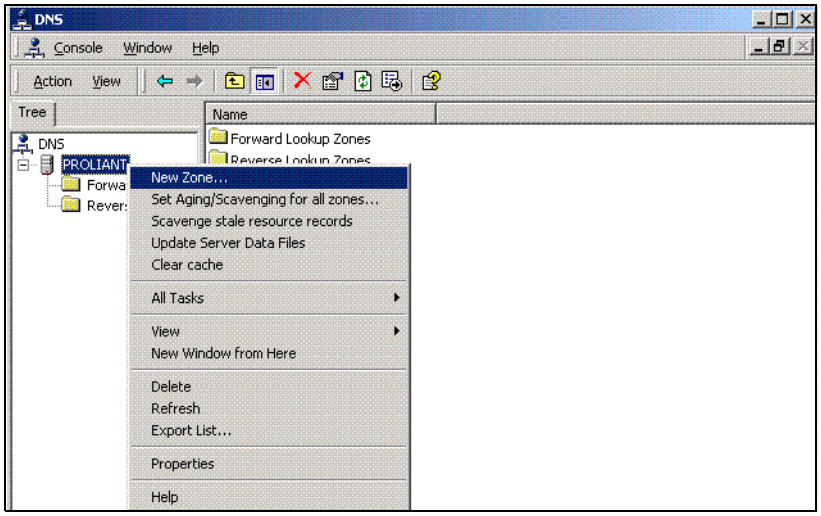


Figure 8-12 Microsoft DNS Manager

[1] Microsoft product screen shots reprinted with permission from Microsoft Corporation.

To create a zone:

a. Select **Standard primary** DNS, as shown in Figure 8-13.

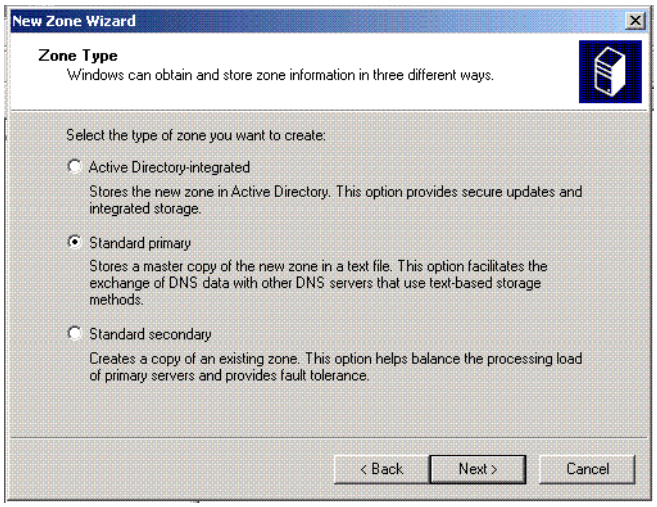


Figure 8-13 Microsoft DNS configuration: Part 1

b. Select **Forward lookup zone**, as shown in Figure 8-14. The whitelist and blacklist lookup need a name to address translation.

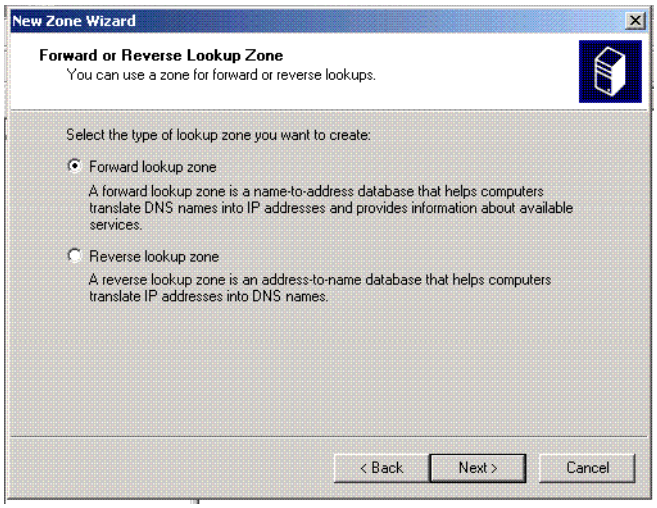


Figure 8-14 Microsoft DNS configuration: Part 2

c. Enter the name of the zone, as shown in Figure 8-15. Our domain is `white.stdi.com`.

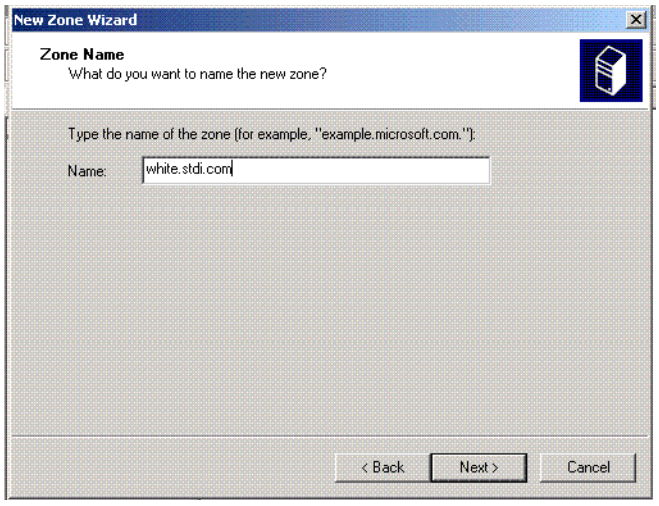


Figure 8-15 Microsoft DNS configuration: Part 3

d. Accept the default for the zone file name, as shown in Figure 8-16.

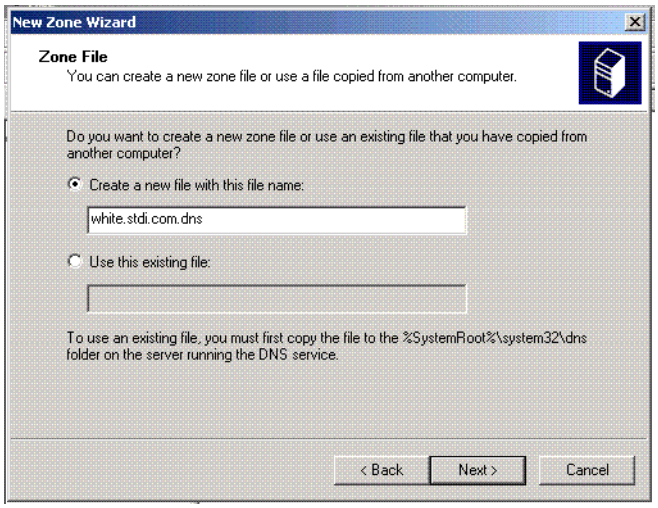


Figure 8-16 Microsoft DNS configuration: Part 4

At this point, the system confirms your answers and we are ready to initialize the DNS. See Figure 8-17.

Figure 8-17 Microsoft DNS configuration: Part 5

3. The first step is to create a placeholder entry in the DNS configuration file. This placeholder makes the first manual changes easier but is not required for the correct working. To create a placeholder:

   a. Double-click the newly created DNS to refresh the entries on the right side of the window. Then, right-click the domain name and select New Host, as shown in Figure 8-18.

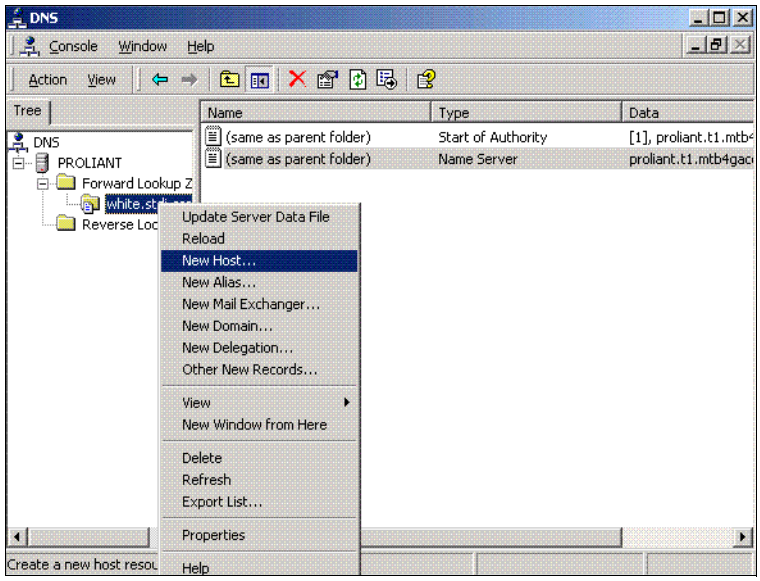


Figure 8-18 Microsoft DNS configuration: Part 6

b. Pick a name as the domain name. Enter the `127.0.0.1` IP address, as shown in Figure 8-19. All your responses to the whitelist and blacklist query should return the 127.0.0.1 address.

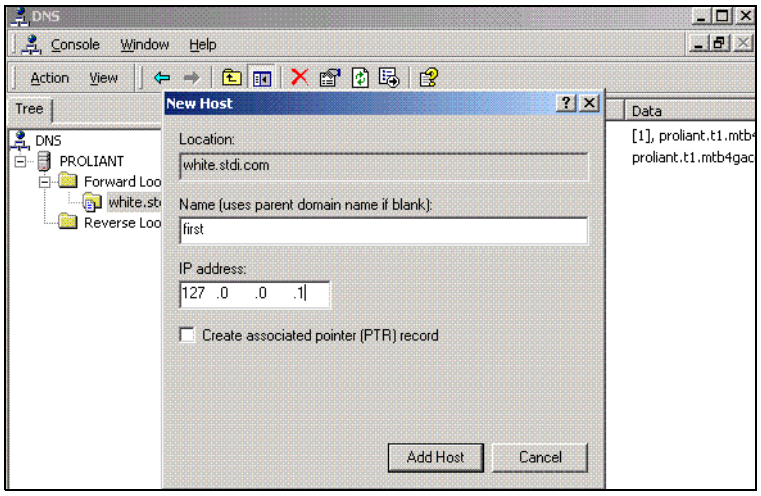


Figure 8-19 Microsoft DNS configuration: Part 7

4. Before we exit the DNS Manager software, highlight the server name and select the **Update Server Data Files**, as shown in Figure 8-20.

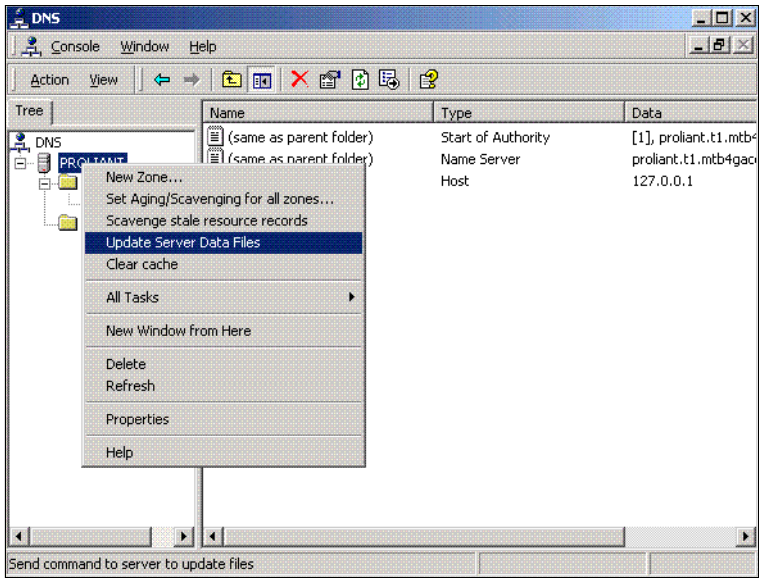


Figure 8-20 Microsoft DNS configuration: Part 8

Exit the software. From this point, we update the DNS configuration file manually.

Zone file updates

Open the DNS configuration file located in the \winnt\system32\dns folder. The file name is the one we selected for the zone file name in step d on page 150. The file looks similar to the one shown in Example 8-8 on page 153.

Example 8-8 Zone file

```
;
;  Database file white.stdi.com.dns for white.stdi.com zone.
;      Zone version:  4
;
@                       IN  SOA white.stdi.com.  admin.white.stdi.com. (
                                4            ; serial number
                                900          ; refresh
                                600          ; retry
                                86400        ; expire
                                3600       ) ; minimum TTL
;
;  Zone NS records
;
@                       NS white.stdi.com.
;
;  Zone records
;
first                   A       127.0.0.1
```

We created the address record for *first*. This entry is now our template to add the whitelist entries.

In this example, we added two IP addresses to our whitelist DNS, as shown in Example 8-9.

Example 8-9 Zone file

```
...
;
;  Zone records
;

first                   A       127.0.0.1
2.1.168.192             A       127.0.0.1
4.156.176.207           A       127.0.0.1
```

Remember to reverse all IP addresses when adding them to the zone file. The IP address 192.168.1.2 becomes 2.1.168.192. Add as many entries as you need.

How do you know if the DNS will work correctly? Use the **nslookup** command to inquire the DNS, as shown in Example 8-10. The **nslookup** command connects to the currently configured DNS.

Example 8-10 nslookup

```
C:\>nslookup
Default Server:  cache02.ca-dns.net
Address:  142.77.2.36

> set root=192.168.1.147
> root
Default Server:  [192.168.1.147]
Address:  192.168.1.147

> first.white.stdi.com
Server:  [192.168.1.147]
Address:  192.168.1.147

Name:    first.white.stdi.com
```

```
Address:  127.0.0.1

> 2.1.168.192.white.stdi.com
Server:  [192.168.1.147]
Address:  192.168.1.147

Name:    2.1.168.192.white.stdi.com
Address:  127.0.0.1

>exit
```

The first step is to verify that the name or address returned by the **nslookup** command is also the one used by the SMTP server. You can change the DNS with the **set root** command. You can verify the connection by entering the **root** command.

Enter the name of the placeholder address, in our example, `first.white.stdi.com`. The server will reply with the IP address 127.0.0.1. Verify any of the other addresses, and the reply should be the same.

8.7 The future of spam

The spam problem we have today is largely the result of non-existent sender authentication. The current SMTP standard does not provide the required protocol. Two new protocols deal with sender authentication and are being reviewed:

- Sender Policy Framework (SPF) uses the existing DNS system to store a list of servers that are authorized to send e-mail for the domain. When an SMTP server receives a message, a simple comparison of the connecting IP address with the listed IP addresses in the DNS will confirm the sending servers identity or not.

  For details, go to:

  http://www.openspf.org

- DomainKeys uses two verifications. First, the sender must be authenticated. Second, the message must pass a scoring system. The scoring system is intended to prevent or slow down large volume e-mail distribution.

  For details, go to the following Web page and click **DomainKeys**:

  http://antispam.yahoo.com

The DomainKeys system is more complex and will take more time to get accepted. SPF, however, is already being used. Anybody can publish an SPF-compliant entry in the DNS without a software upgrade. As a result of the simplicity, it is already used by large providers. It is possible that both standards will get accepted.

A

Notes C API security enhancements

Lotus Notes/Domino 7 unveils a series of new Lotus Notes C API calls that manipulate aspects of the distribution of Notes ID files and their use in securing Notes documents such as e-mail messages. For a Lotus Developer Domain article that discusses the APIs thoroughly and how they can be functionally implemented, see “Security APIs in Notes/Domino 7.0,” available at the Lotus Technical Library area of:

`http://www.lotus.com/ldd`

We keep our discussion at a high level and suggest scenarios where the APIs might prove useful.

The names of the APIs suggest their purpose[1]:

- `SECKFMOpen()`, `SECKFMClose()`
- `SECAttachIdFileToDB()`, `SECExtractIdFileFromDB()`
- `SECRefreshIDFile()`
- `NSFNoteSignExt3()`, `NSFNoteCopyAndEncryptExt2()`, `NSFNoteDecryptExt2()`

`SECKFMOpen()`, unlike the `SECKFMSwitchToIDFile()` call introduced in Notes 5, allows an ID to be unlocked by means of a password without forcing the entire Notes execution context over to that ID. The call emits a context for use with the content security calls previously listed (encrypt, sign, decrypt). Therefore, now a Notes C API program can seamlessly perform cryptographic operations with credentials other than those governing the current Notes execution context.

`SECAttachIdFileToDB()` and its counterpart allow ID files to be distributed beyond the immediate file system. The calls are careful to do this securely by implementing a variant of the Bellovin-Merritt Encrypted Key Exchange protocol for secure transmission of keys to and from a server. In practice, this involves double encryption such that the Domino database

[1] The prefix SEC is short for “security”; KFM is an acronym for “Key File Management”; NSF is, of course, an acronym for “Notes Storage Facility.”

housing the ID attachment does not possess as such the secret password needed to decrypt the attachment into a usable state.

`SECRefreshIDFile()` causes an ID file to be invested with any automatic updates slated for it at the Domino server. This can include X.509 certificate updates, Notes recertifications (name changes, expiration extensions), and key rollovers. therefore, an ID maintained apart from the user can be programmatically kept up to date.

One can imagine how this array of functions might be leveraged by IBM Business Partners, independent software vendors (ISVs), and so forth:

- Applications that handle secured content on behalf of users in new ways, enabling it, for example, to be sent to and processed by other systems or applications, such as e-mail to a personal digital assistant (PDA) device. Domino Web Access exemplifies this, carrying out cryptographic operations on e-mail by means of imported ID files. (See 7.5, “Secure messaging with Domino Web Access” on page 117.)
- Applications that allow distributed indexing and archival of secured Notes mail.
- Applications that manage Notes IDs in ways different from what Domino ID recovery provides (a different central managed repository, for example).
- One-off applications in specialized circumstances where cryptographic security and manipulations are useful.

B

Quick server security checklist

There are a number of things that should be done immediately to any new Domino server whenever it is installed in order to lock it down and make it ready for secure use. This is especially true when configuring the initial server in a new Domino domain. This appendix provides a checklist of tasks to do or verify. It is not necessarily exhaustive for all possible server configurations, but you can use it as a quick guide for the most essential steps.

Locking down the directory

In this section, we describe how to lock down the directory.

Securing the Internet password hash

Since Release 4.6, Domino has supported two different algorithms (known as “hashes”) for storing users’ Internet passwords. One is fairly weak and potentially subject to dictionary attacks, but it is backward compatible all the way to Release 4.5. The other uses a technique called “salting” to make dictionary attacks impractical.

Due to concerns about backward compatibility with Release 4.5, the weak, unsalted method is still the default. If you have 4.5 servers still in your configuration, you cannot use the newer and stronger method. (We highly recommend that you upgrade those servers.) Otherwise, one of the first things you should do when you have the first server in a domain up and running is to enable the stronger password. To do this, follow these steps:

1. Open the Domino Directory (NAMES.NSF) in either the Notes client or the Administrator client.
2. Select **Edit Directory Profile** from the Actions menu.
3. Set the Use more secure Internet Passwords field to **Yes**.

Important: If you are setting up a server in an existing domain, it is still a good idea to check this setting. If it is not set to “Yes,” change it immediately. In addition, go to the People view, select all the documents and then select **Upgrade to More Secure Internet Password** from the Actions menu.

Setting the ACL

The exact configuration of the ACL for your Domino Directory can vary, but it is a good idea to lock it down completely first, and then loosen settings as needed. We recommend the following steps:

1. Make sure that the Default and Anonymous entries in the ACL are set to **No Access**.
2. Make sure that the Maximum Internet name and password access level is set to **Reader** unless you intend to use the Web Administrator client, in which case, set it to **Editor**.
3. There are several roles listed in the ACL of the Domino Directory. These roles can be used to delegate specific responsibilities for tasks such as managing Person documents, Group documents, and Policy documents. If you expect to be sharing administration tasks with other administrators and you want to limit what things they can do, create a series of Groups, add them to the ACL, assign them the predefined roles, and add people to the Group documents.

Setting permissions in the Server document

In this section, we describe how to set permissions in the Server document.

Essential settings

The Security tab in the Server document contains a number of fields that control important permissions. Some of these fields should always be set immediately. The importance of some

of the others varies depending on what you are going to do with your server. Always complete the following actions:

- Develop a standard for group names that you can use consistently in these fields for all your servers. You might want to consider a convention that differentiates people who have responsibilities across all your servers from people who have responsibilities only on individual servers. For example, two such group names might be “Acme Corp Domain Admins” and “Acme Corp Server01 Admins.”
- Enter group names into at least the following fields in the Server document:
  - Full Access Administrators
  - Administrators
  - Not Access Server
  - Create Databases and Templates
  - Create New Replicas
  - Create Master Templates
- Add appropriate names to the groups.

Important: Permissions to create databases, new replicas, and master templates have significant security implications. They are not just there to keep unauthorized users from cluttering up your server and consuming disk space.

Review other Server document settings

All the settings on the Security tab are significant and should be reviewed. Full descriptions of all of them are beyond the scope of this appendix. Refer to the Lotus Domino 7 Administrator Help database for more information.

Make your templates secure

Every Domino server has a large number of database template files (NTF files). The installation process for the server simply copies those files to your disk as-is. The server setup process can automatically set the anonymous entry in template ACLs to “No Access,” but this might not be sufficient protection in all cases. Every Domino server in the world potentially has templates that can replicate their customized designs into your templates. To prevent this, it is a good idea to change the replica IDs in your templates. There is no easy way to do this with the Domino Administrator client, but a free tool called “Surely Template” makes this very easy. The tool is available at:

http://www.openntf.org

C

Domino as a certificate authority

In the beginning, Lotus Notes served as both the client and server. With the advent of the Internet, there was a need to extend and embrace Internet security functionalities and standards. This is when Notes came to identify the client and Domino came to be the name of the server. Back then, changing the name was more than cosmetic; it hinted at the fact that the Domino server was now able to serve both Notes and Web browser clients. With Domino 7.0, the Domino server continues to fulfill that role.

Of interest to us in earlier chapters, and in this appendix, are Internet security services provided by Secure Sockets Layer (SSL) and Secure/Multipurpose Internet Mail Extensions (S/MIME). SSL is responsible for ensuring the secure encryption of the communication channel between the Web browser and the server and certificate-based Internet authentication; the S/MIME is responsible for the exchange of encrypted and digitally signed e-mails over SMTP.

You can set up SSL on a Domino server so that clients and servers that connect to the server use SSL to ensure privacy and authentication in the network. SSL is set up on a protocol-by-protocol basis. For example, you can enable SSL for mail protocols -- such as IMAP, POP3, and SMTP, and not for other protocols.

To set up SSL on the server, a key ring containing a server certificate from an Internet certificate authority is required. You can request and obtain a server certificate from either a Domino or third-party certificate authority (CA) and then install it in a key ring. A server certificate is a binary file that uniquely identifies the server. The server certificate is stored on the server's hard drive and contains a public key, a name, an expiration date, and a digital signature. The key ring also contains root certificates used by the server to make trust decisions.

Regarding S/MIME, before Internet and Notes clients can use client authentication or send signed mail, they must have an Internet certificate. To send encrypted mail using S/MIME, they must have the recipient's Internet certificate.

This requires the use of the Domino Certificate Authority for Internet and Notes clients that are creating new public and private keys for the Internet certificate. This is not needed if the user is using a Notes client and the CA issued certificates in the Person document of the

Domino Directory. Notes automatically adds Internet certificates stored in the Person document to the Notes ID file when the user authenticates with the server.

You can also set up Notes clients to use different certificates for signing and encryption so that it is possible to designate one Internet certificate authentication and signing and another for encryption.

Therefore, the purpose of this appendix is to explain the tools available, what must be done to best use them, and details of how to implement correctly these Internet security services.

In other words, we look at the necessary steps to create a Domino-based CA, server certificates, and client certificates to support SSLv3 (x.509v3) certificate-based authentication. The client certificates can then, in turn, be used for the purpose of sending and receiving secure Internet messaging, using the Secure/Multipurpose Internet Mail Extensions (S/MIME).

We describe the following main topics:

- Creating the Domino Certificate Authority
- Requesting and installing a server certificate
- Requesting, picking up, and using a client certificate

We will cover each section succinctly, specifically describing each step of the operations required to accomplish each task correctly.

Finally, before we start, we need to clarify an important detail regarding Domino and the certificate authority, because there are two of CAs.

There is the server-based Domino certificate authority (also called the CA process), which is a Domino server task that is used to manage and process certificate requests. The CA process runs as an automated process on Domino servers that are used to issue certificates. A Notes or Internet certifier is linked to the CA process on the server in order to take advantage of CA process activities. Only one instance of the CA process can run on a server; however, the process can be linked to multiple certifiers.

We do not cover the CA process here and how some of its features applies to Notes IDs. Instead, we recommend that you consult the concise and well written document *The CA Process In Notes/Domino 6* (which applies to Notes and Domino 7), available at:

`http://www.ibm.com/support/docview.wss?rs=463&uid=swg27006424`

Then, there is also the Domino 5 Certificate Authority, which describes the Domino Certificate Authority application on the server using the Domino R5 Certificate Authority template (CCA50.NTF).

This is what we cover in this appendix to ensure that the information required to perform some of the tasks explained in other sections of this book can be understood and performed without having to look elsewhere for information.

Creating the Domino Certificate Authority

The Domino CA application has the following functions:

1. Create and manage the Domino CA key ring file, which holds the certificate authority SSL certificate.
2. Sign (that is, certify) server and client public keys when requested to create new certificates.

   To do so, the server administrator or client must paste a request into the Certificate Authority database to start the certification process. The Domino CA application enables the automatic or manual addition of client certificates created by the application to be added to the Domino Directory.
3. Add client certificates to the Domino Directory on request.

   The client certificates can be either from the Domino CA (if necessary) or an external certificate authority. Note that, even if you choose to use an external certificate authority to create server and client certificates, you can choose to install the CA application for the purpose of adding client certificates to the Domino Directory for authentication.
4. Create a server key ring file with a signed server certificate for the server on which the Domino CA is installed.

   This optional feature is a convenience to the CA administrator, but is not absolutely necessary. In this document, we illustrate requesting server certificates through a Web browser because this would be necessary in any case for any Domino server other than the CA server and if using an external certificate authority.

In order to create the Domino CA, you must create the Domino Certificate Authority database. This permits the creation of an SSL certifier. It is important to note that the SSL certifier can only sign certificate requests, and, once a server's or client's public key (plus distinguished name and other identifying material) is signed by the SSL certifier, it can be used as an SSL certificate. It is worth also noting that server or client certificates are used for SSL session setup and authentication, SSL certifiers are exclusively used to certify other public keys, and clients can also use the certificates for S/MIME encryption and signing e-mail (if they supply their e-mail address to be included in their certificate).

Step 1: Create the Certificate Authority database

Select **File** → **Database** → **New**, which opens the New Database dialog box, as shown in Figure C-1 on page 164. Enter the example information (this is an example only; you can enter information that is more appropriate to your particular setup). Note that it will be necessary to show the Advanced Templates to access the Domino Certificate Authority (6) template.

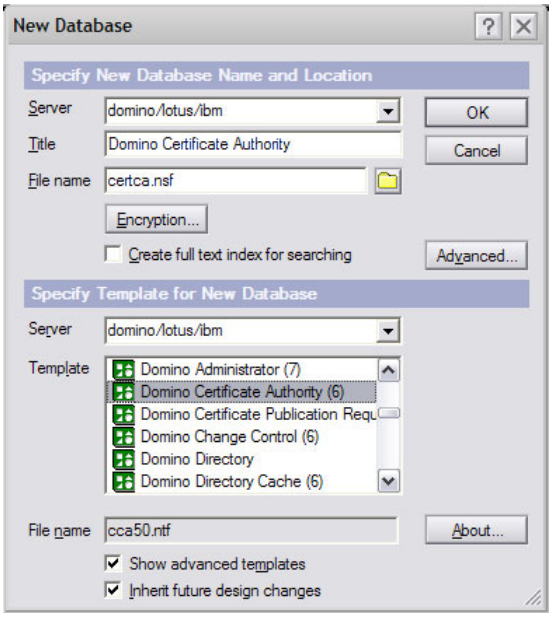


Figure C-1 New Database dialog box

Exit the database and ensure that the account being used is represented in the ACL, including having the [CAPrivilegedUser] role selected.

Step 2: Create the certificate authority key ring and certificate

Go back in the Domino Certificate Authority database (we perform two of the three steps in the Domino CA menu) and select **1. Create Certificate Authority Key Ring and Certificate**. This opens the Create Certificate Authority Key Ring form, as shown in Figure C-2.

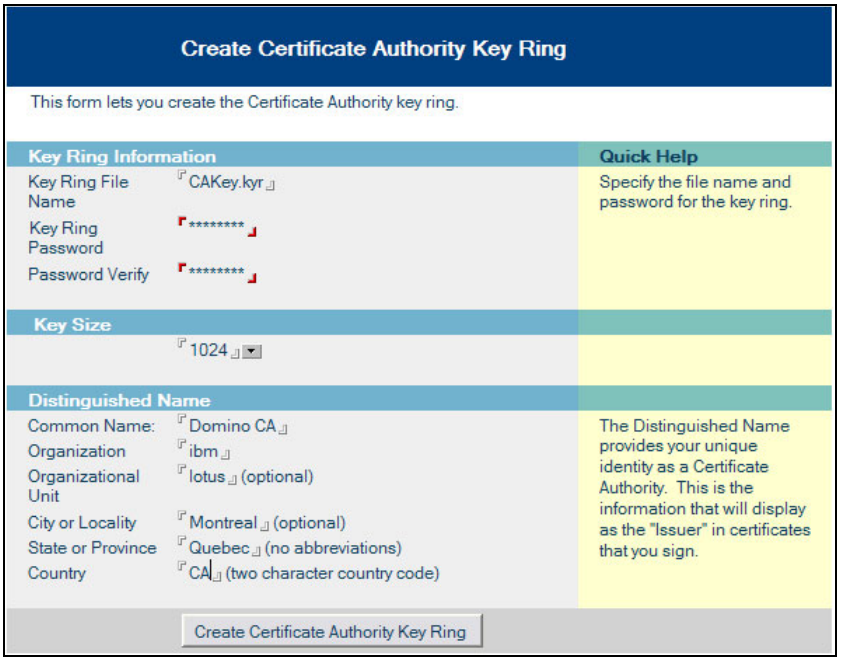


Figure C-2 The Create Certificate Authority Key Ring form

Note that the key ring file is created in the c:\notes\data directory. After clicking the **Create Certificate Authority Key Ring**, a confirmation dialog box opens, as shown in Figure C-3.

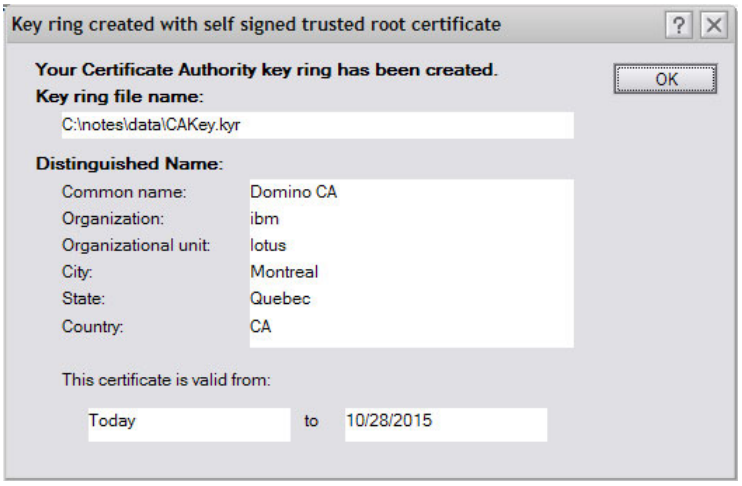


Figure C-3 The Certificate Authority key ring confirmation dialog box

Step 3: Configure the certificate authority profile

The purpose of this profile is to create a set of default values to be used when signing certificate requests from this certificate authority. Select **2. Configure Certificate Authority Profile**. This opens the Certificate Authority Profile form, as shown in Figure C-4 on page 166.

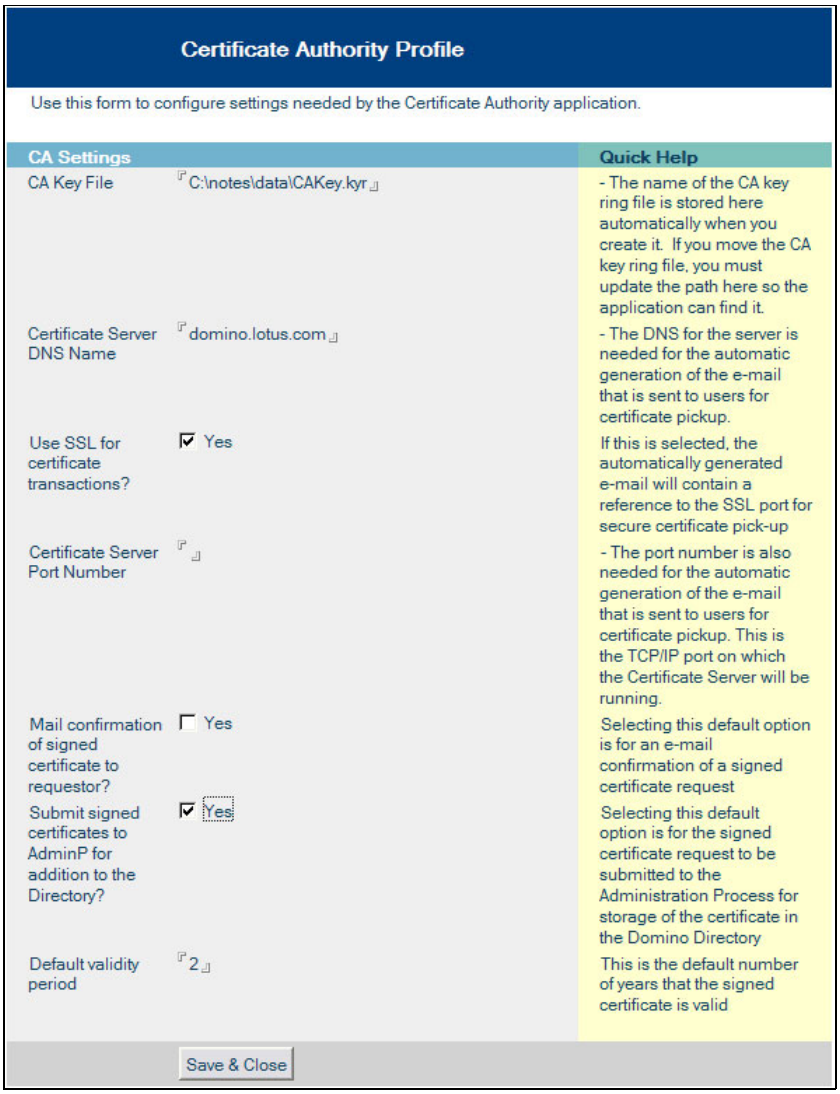


Figure C-4 The Certificate Authority Profile form

Enter the following information:

- CA Key File: `c:\notes\data\CAKey.kyr`

  This defaults to the location where you originally created the certificate authority key ring file. If you moved it since creating it, point to the new location where you will keep the key ring file permanently.

- Certificate Server DNS Name: `domino.lotus.com`

  This is necessary to create the host name part of a reference URL in confirmation e-mails. If you choose not to inform users by e-mail, you do not need to fill in this field.

- Use SSL for certificate transactions: **Yes**

  This is necessary to allow the URL in a confirmation e-mail to be set up properly (HTTPS rather than HTTP). This setting changes the protocol from “HTTP” to “HTTPS.” Note that, if you select this option, you must configure the Domino server on which the CA is installed to use SSL by creating and installing a server certificate and activating SSL. We describe this in “Requesting and installing a server certificate” on page 167.

- SSL Certificate Port Number: <blank>

  This is used to configure the URL in the confirmation e-mail. If you plan to change this in the server configuration, use the same HTTPS SSL port number. Note that the port number will not show in the URL created in the e-mail if it is the default (443) value.

- Mail confirmation of signed certificate to requestor: **Yes**

  Select this if you want automated confirmation to be sent when you approve certificates.

- Submit signed certificates to adminp for addition to the Directory: **Yes**

  This automatically adds any client certificate you approve to the Domino Directory by submitting a request to the Domino Administration Process. Note that, if the CA server is not also the administration server for the Domino Directory, the request will have to be replicated to that server to be processed. If you chose not to have this done automatically, you can still manually initiate the request from the document containing the approved certificate subsequently.

- Default validity period: 2

  The default of two years is adequate for most client certificates; you can change this when approving a certificate. Note that the validity period begins from 12:00 midnight UT on the day a certificate is approved, not from the moment of approval. Therefore, the validity period can begin before or after the moment of approval depending on your local time zone.

Scroll down and click **Save & Close** to return to the certificate authority main menu. Note that the options you chose will be stored in the certificate authority, so you will not be prompted to enter the certificate authority's key ring file password.

Step 4: Create the server key ring and certificate

There is a third option on the menu that enables you to create a server key ring with a certificate from the Domino CA on the same server as the certificate authority. This is convenient if you have a single test server, but is not applicable to multiple servers or if obtaining certificates from an external certificate authority. The next steps show how to get server certificates from a certificate authority (either a Domino or external certificate authority) using a browser and the Domino Server Certificate Administration database to manage the server's key ring file.

Therefore, we do not perform this step as part of the current procedure.

Requesting and installing a server certificate

This is also know as Domino Server Certificate Administration database creation and configuration.

To enable SSL on a server, it must have access to a key ring file containing the server's certified public key, private key, and one or more certificates flagged as "trusted roots" from certificate authorities that have certified the server's public key. This will later enable the creation of trust relationships using SSL certificates between browsers and servers and between servers using SSL.

The Domino Server Certificate Administration database is necessary to manage server certificates whether you use the Domino certificate authority (which we use in our testing) or an external certificate authority. You need to create this database before requesting a certificate.

Step 1: Create the Server Certificate Administration database

Select **File** → **Database** → **New**, which opens the New Database dialog box, as shown in Figure C-5. Enter the example information (this is an example only; you can enter information that is more appropriate to your particular setup). Note that it will be necessary to show the Advanced Templates to access the Domino Certificate Authority template.

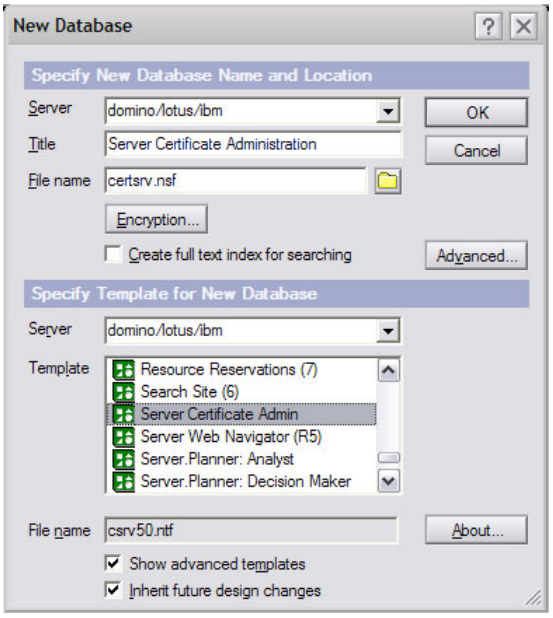


Figure C-5 Creating a database

Exit the database and ensure that the account being used is represented in the ACL (unlike the Certificate Authority database, there is no special role to be mindful of here).

Step 2: Create the certificate authority key ring and certificate

Go back in the Server Certificate Administration database. Before we perform the needed steps, note that “Step 4: Create the server key ring and certificate” on page 167 is for testing purposes where only a server (site) “self-signed” certificate is required (not signed by a known certificate authority). Because we might want to use client certificates for authentication, this option is not appropriate for our purposes. In addition, it is intended for testing purposes (using SSLv2) only and should *not* be used in a production environment.

Here, we perform the following four steps in the Server Certificate Administration menu:

1. Create the key ring.

   This creates the server’s key ring file plus a “stash” file (with an .sth extension) to hold the key ring’s password.

2. Create the certificate request.

   This creates a key pair (public and private) and also displays the public key in PKCS-12 format to allow it to be pasted into a Web browser for submission to a certificate authority.

3. Install a certificate authority’s certificate (flagged as “trusted root”) into the key ring.

   This is necessary to set up a “chain” of certificates from the trusted root certificate authority’s certificate to the server’s certificate. Do this before installing the server’s certificate into the key ring file.

4. Install the certificate into the key ring.

   After the certificate request from step 2 has been approved, it is “picked up” in a browser and added to the server’s key ring file.

Select **1. Create Key Ring** from the main menu of the Server Certificate Administration database. This opens the Create Key Ring form, as shown in Figure C-6. Enter the information shown in the figure.

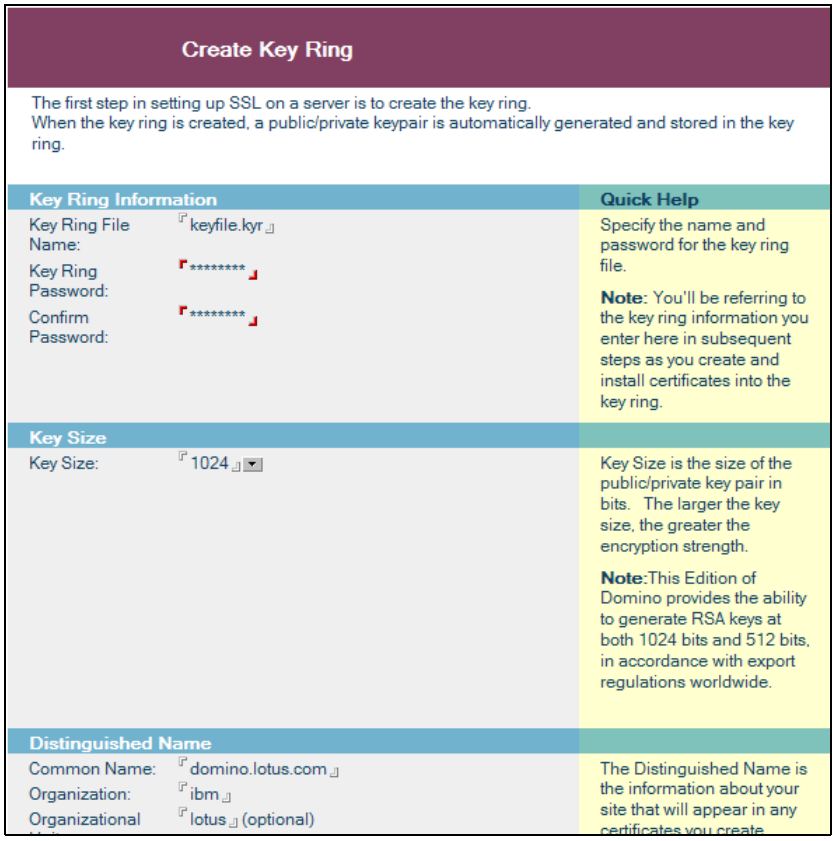


Figure C-6 The Create Key Ring form

Note that if the Key Ring File Name is set to anything other than `keyring.kyr`, it requires that the name is changed in the server’s Server document (in the Ports → Internet Ports tab). This is because this field also defaults to the “keyring.kyr” name.

Click **Create Key Ring**, and the key ring file with a public-private key pair will be created and stored on a local disk of the workstation from which the request was submitted, and not the target server. A confirmation dialog box similar to the one shown in Figure C-7 on page 170 opens.

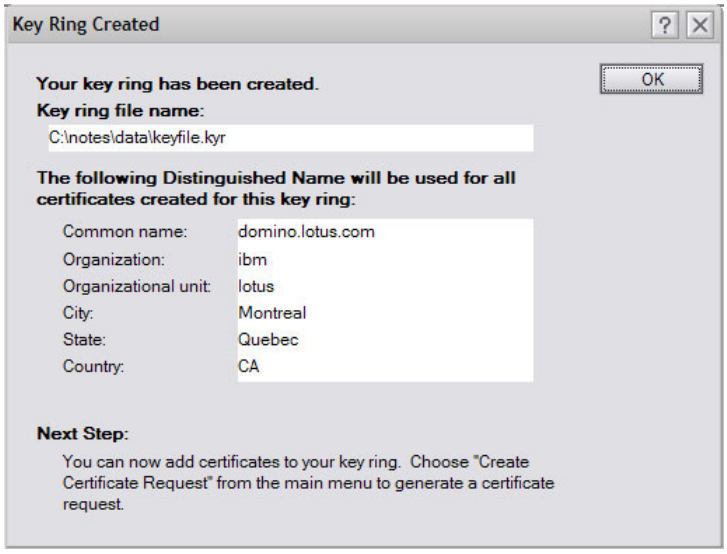


Figure C-7 The Key Ring Created confirmation dialog box

We now have a public-private key pair in a key ring file. Because the public key is unsigned, the next step is to request a certificate authority to certify the public key. (The private key never leaves the key ring file).

Step 3: Request a server certificate from a certificate authority

Perform the following steps:

1. Select **2 - Create Certificate Request**. This opens the Create Server Certificate Request form, as shown in Figure C-8 on page 171. Enter the following information:
   - Key Ring File Name: `c:\notes\data\keyfile.kyr`

     Confirm that the correct key ring file is filled in; by default, this will be the key ring file just created, but you can substitute another if wanted.
   - Log Certificate Request: **Yes**

     If Yes is selected, a copy of the request will be kept in the Server Certificate Administration database for later reference.
   - Method: **Paste into form on CA's site**

     You can chose to paste the server's SSL public key into a form on the CA site or to e-mail it to the CA site. Most CA sites prefer that you paste the key into a form on their site.

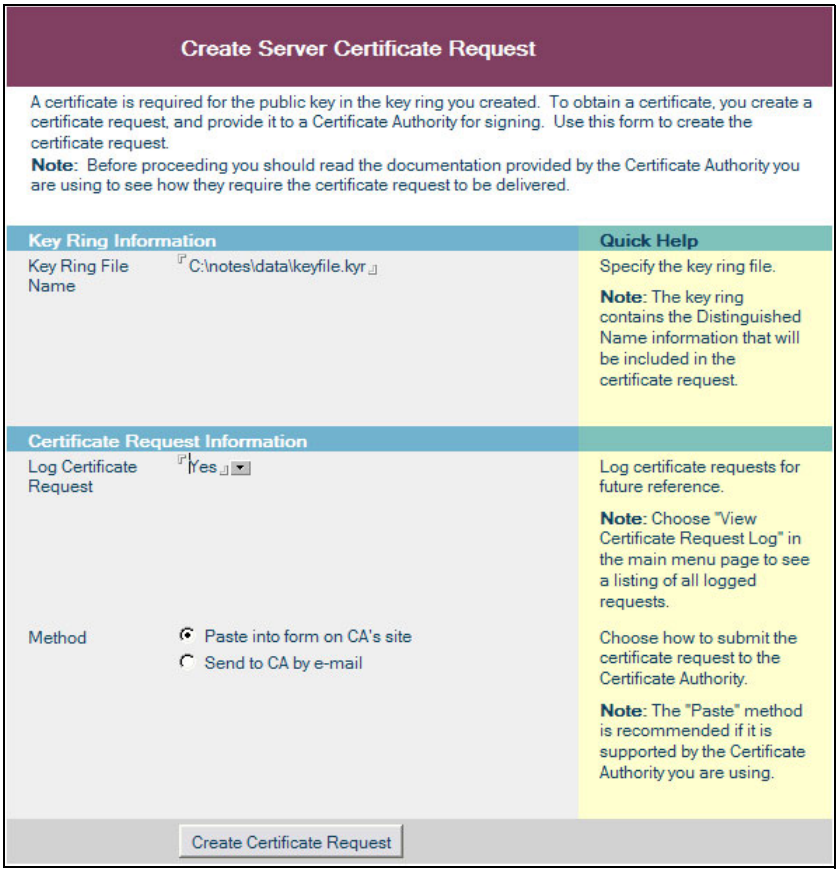


Figure C-8 The Create Server Certificate Request form

2. Click **Create Certificate Request**. The server key ring password prompt dialog box opens, as shown in Figure C-9.

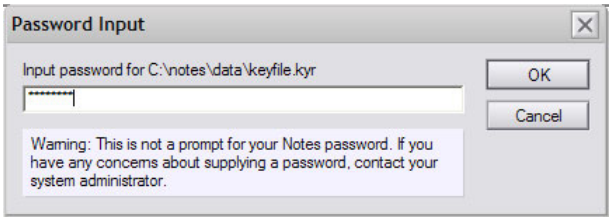


Figure C-9 The Password Input dialog box

3. When prompted, enter the server key ring file's password and click **OK**. A confirmation dialog box opens, as shown in Figure C-10 on page 172.

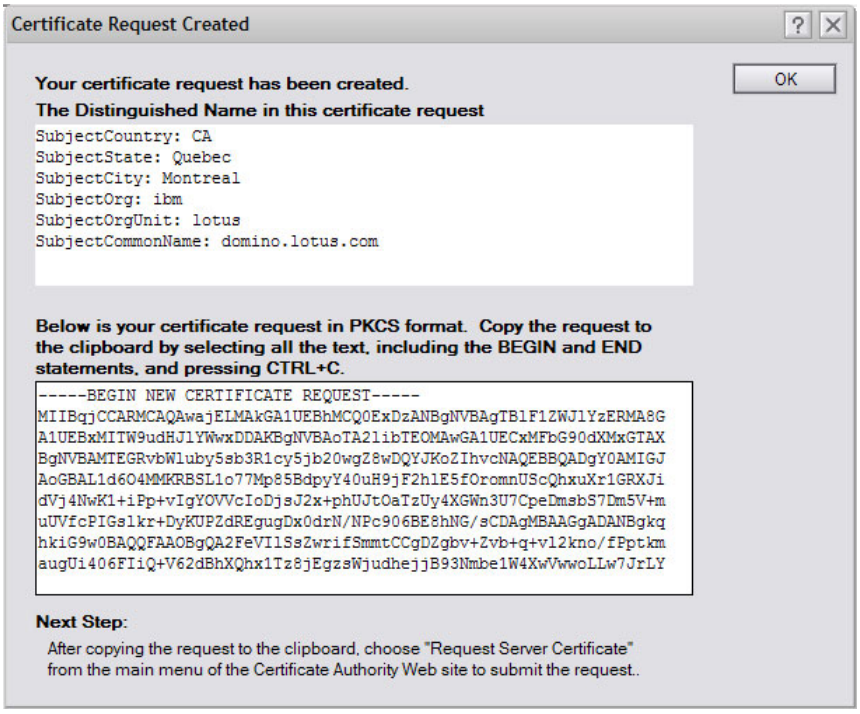


Figure C-10 The Certificate Request Created confirmation dialog box

4. In the Certificate Request Created confirmation dialog box, select all of the text in the bottom pane and copy it to the Clipboard (you might need to scroll the display to capture all the text). Note that it is a good practice to paste it to an instance of Notepad in case something requiring the Clipboard is done between this step and the next step. In addition, make sure to include the “BEGIN NEW CERTIFICATE REQUEST” and “END NEW CERTIFICATE REQUEST” lines for the request to be valid.

   You can now request a server certificate from a certificate authority, which needs to be done through a Web browser.

5. With a Web browser, navigate to the Domino Certificate Authority database (CERTCA.NSF, the name we gave it earlier) by typing in the appropriate URL for your server and CA file name (for example, `http://domino.lotus.com/certca.nsf`). The navigator of the Domino CA opens, as shown in Figure C-11.

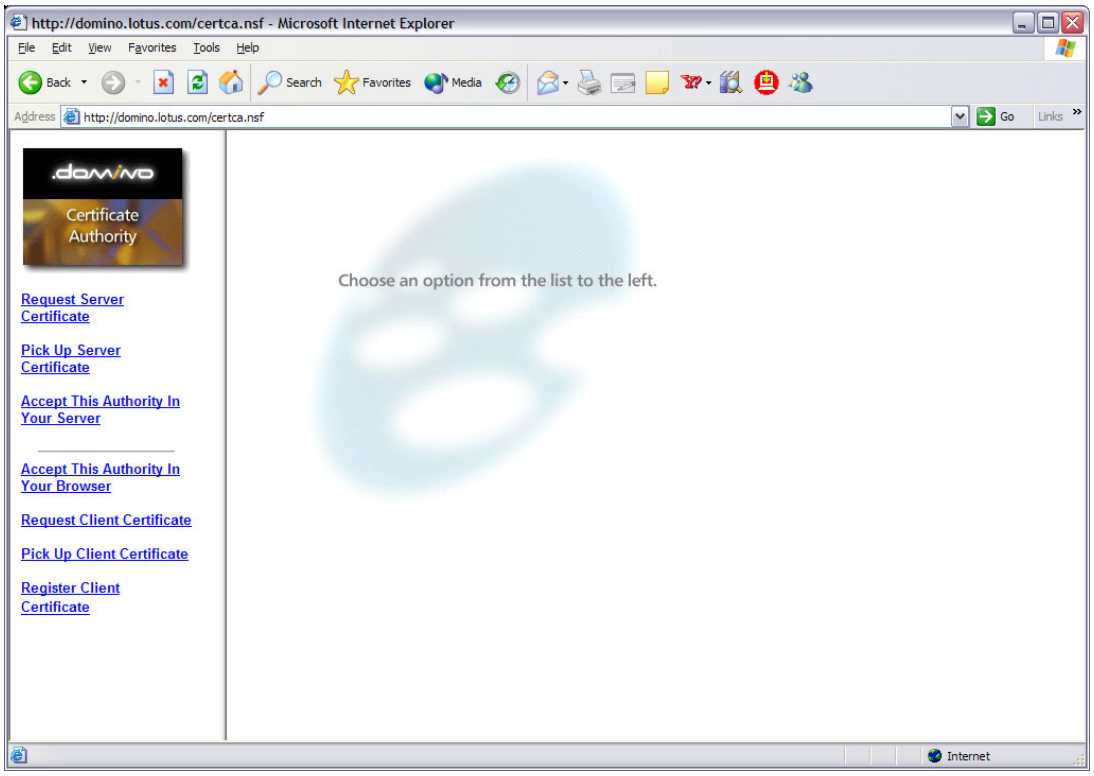


Figure C-11 The Domino Certificate Authority database shown in the a Web browser

6. Select **Request Server Certificate** from the menu on the left. The Request a Server Certificate form opens, as shown in Figure C-12. Enter the information as shown in the form.

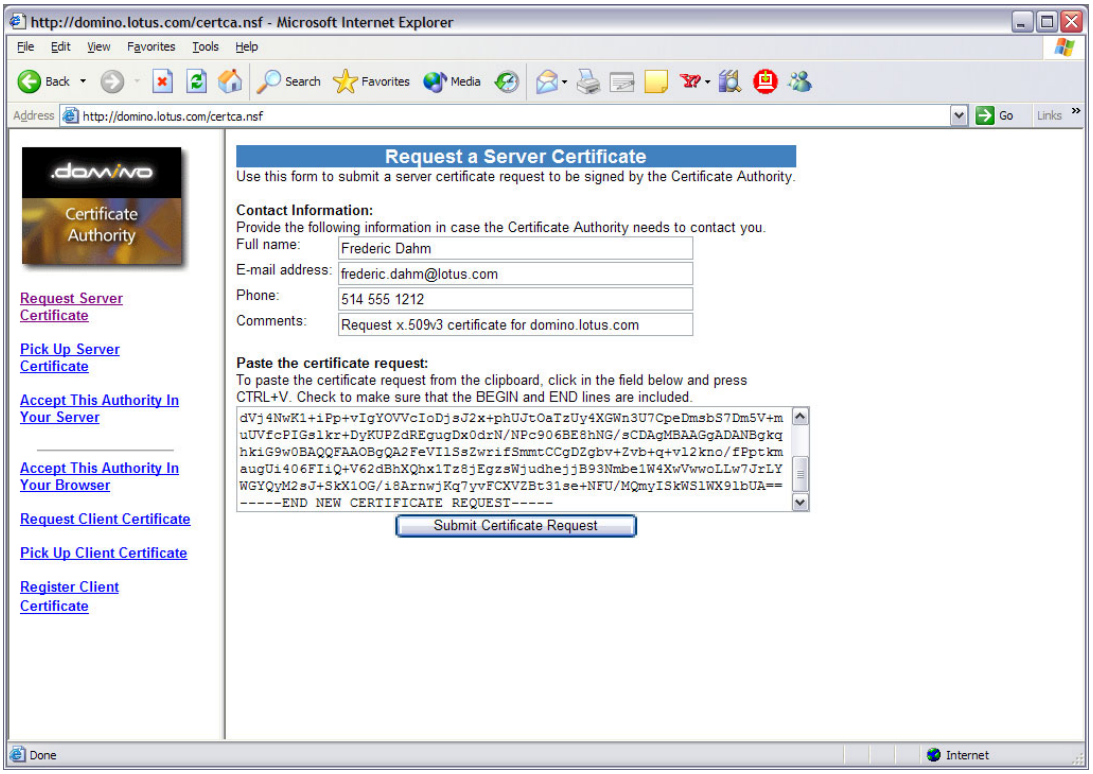


Figure C-12 The Request a Server Certificate form

7. After entering all the information, click **Submit Certificate Request**. The Certificate Request confirmation form opens, as shown in Figure C-13.

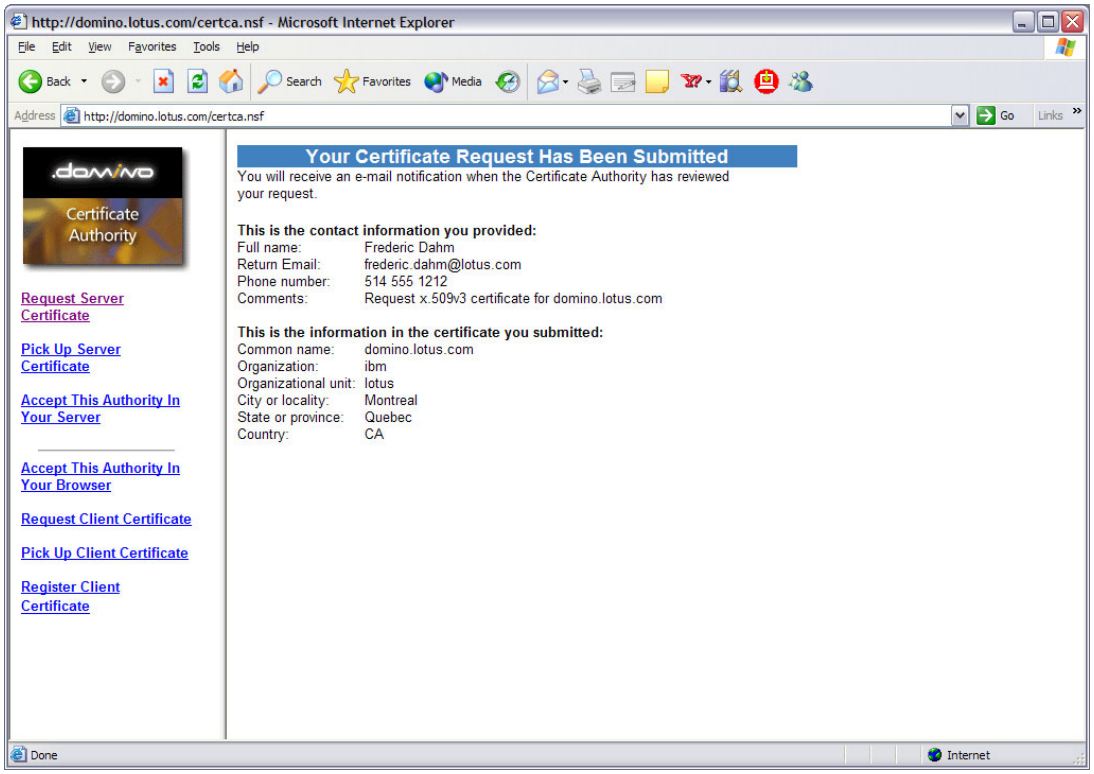


Figure C-13 The Certificate Request confirmation form

We are now ready to merge the certificate authority's certificate into the server key ring.

Step 4: Merge the certificate authority's certificate into the server key ring

This step is necessary to build a chain of trust starting at the certificate authority's root certifier. By doing this, you are setting up a trust relationship: Your server will accept any certificates signed by this CA.

Perform the following steps:

1. Click **Accept This Authority In Your Server** in the Domino Certificate Authority database (still through the Web browser). This opens the Pick Up Certificate Authority Trusted Root Certificate document, as shown in Figure C-14 on page 176. The contents of the document permits to you to install this in your server's key ring file flagged as a "trusted root."

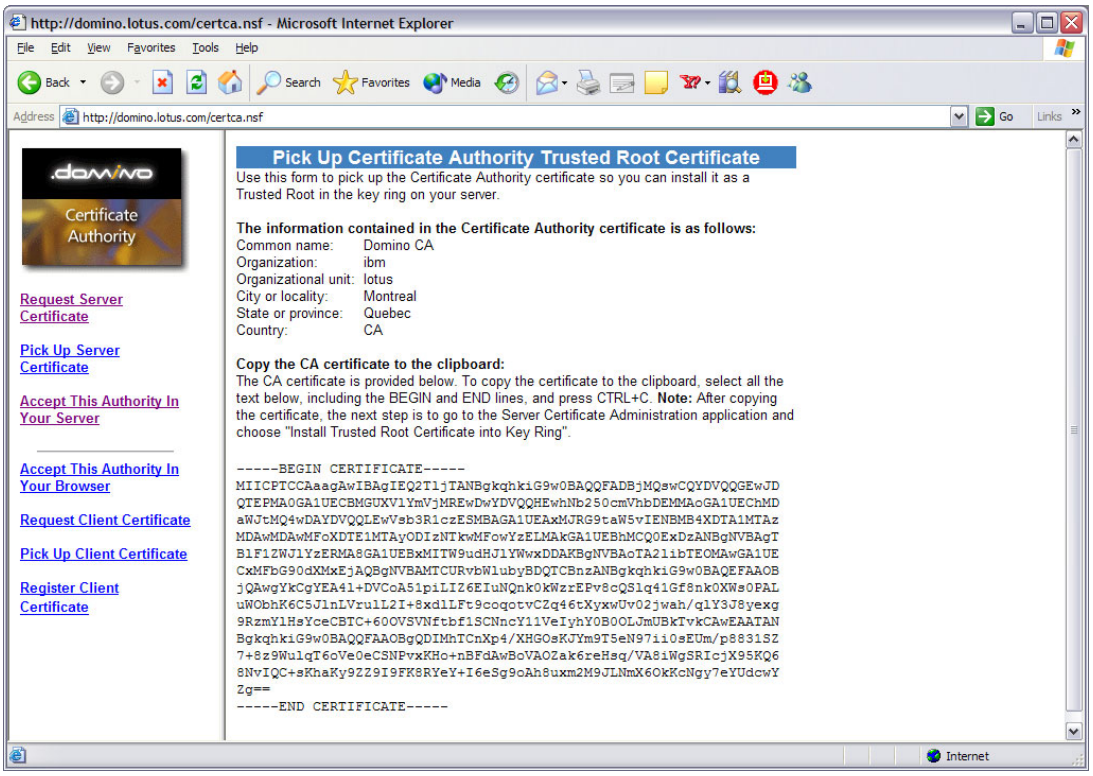


Figure C-14 The Pick Up Certificate Authority Trusted Root Certificate document

2. In the document, select the certificate information on the browser panel and copy the certificate to the Clipboard. Be sure to select the two lines “BEGIN CERTIFICATE” and “END CERTIFICATE”, as well as the certificate text. Note that it is a good best practice paste it to an instance of Notepad in case something requiring the Clipboard is done between this step and the next step.
3. Open the Server Certificate Administration database (CERTSRV.NSF) and select **3 - Install Trusted Root Certificate into Key Ring**. The Install Trusted Root Certificate form opens, as shown in Figure C-15 on page 177. Note that the key ring file in the Key Ring File Name field is the one for the server. Also note also that the key ring file must be on (or accessible to) the workstation submitting the request.

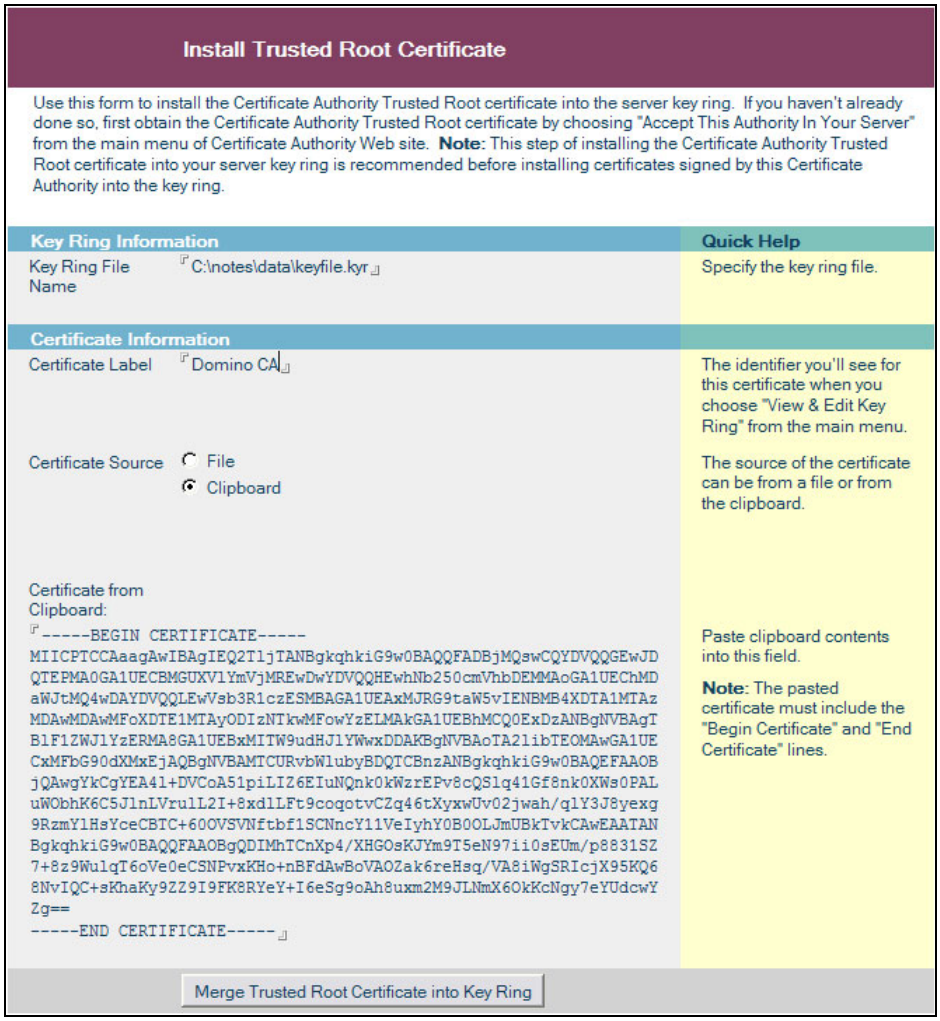


Figure C-15 The Install Trusted Root Certificate form

4. Click **Merge Trusted Root Certificate into Key Ring** at the bottom of the panel. This opens the server key ring password prompt dialog box, as shown in Figure C-16.

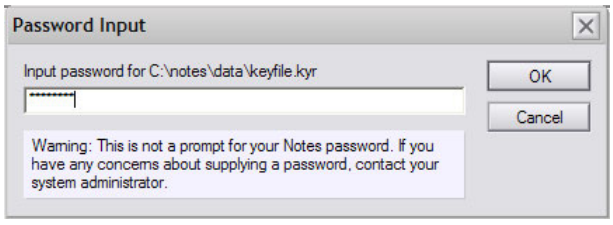


Figure C-16 The server key ring password prompt dialog box

5. When prompted, enter the server key ring file's password and click **OK**. A confirmation dialog box opens, as shown in Figure C-17.

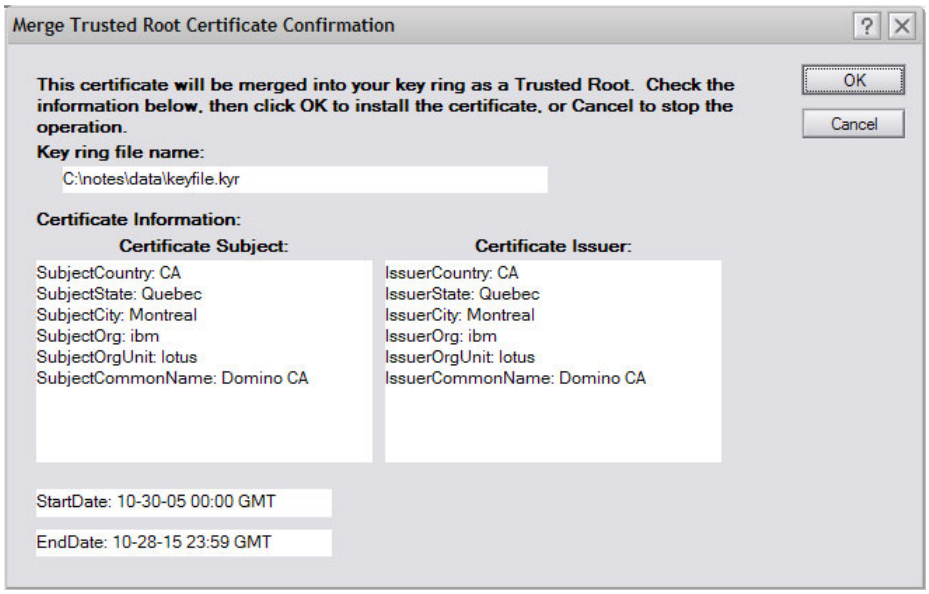


Figure C-17 The Merge Trusted Root Certificate Confirmation dialog box

6. Click **OK**. the Certificate received into key ring and designated trusted root message box opens, as shown in Figure C-18.

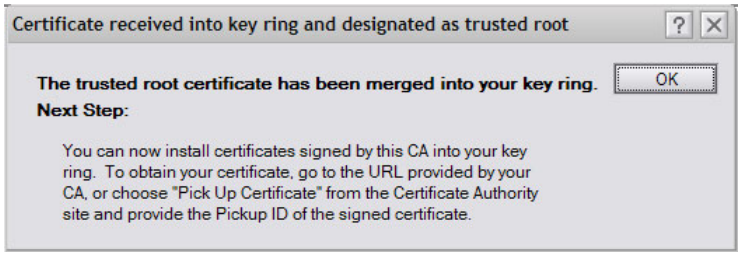


Figure C-18 The Certificate received into key ring and designated trusted root message box

7. Click **OK** to close the message box after the trusted root certificate is merged into the key ring.

The certificate authority trusted root certificate has now been merged into your key ring. You can examine it and other certificates in your key ring and remove the "trusted root" designation or delete it at a future date by selecting **View and edit key rings** from the Server Certificate Administration database main menu. This completes the addition of the CA trusted root certifier to the server's key ring.

Step 5: Install the certificate into the key ring

In general, this request would be to a external certificate authority whose processes would not be visible to the requestor. Because we are using the Domino Certificate Authority, we also show the steps to approve a server certificate request. The subsequent process to "pick up" the certificate in a browser is same whether using the Domino CA or an external CA.

Perform the following steps:

1. First, we need to approve the server certificate request in the Domino CA:

   a. Using the Notes client, open the Domino Certificate Authority database (CERTCA.NSF) to the main menu. Select **Server Certificate Requests** from the menu on the left pane. The Server Certificate Requests\Waiting for Approval view opens, as shown in Figure C-19.

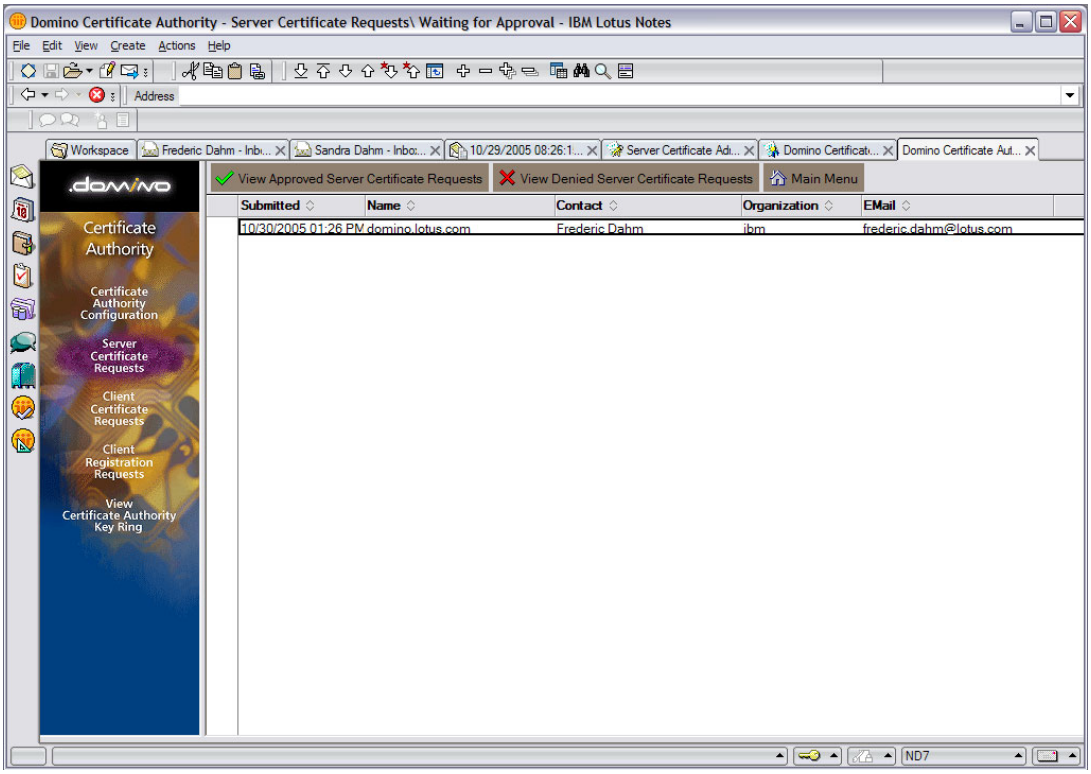


Figure C-19 The Server Certificate Requests\Waiting for Approval view

   b. From this view, select the server certification request (there is only one in our example) and open the document. The Certificate Request Approval form opens, as shown in Figure C-20 on page 180.

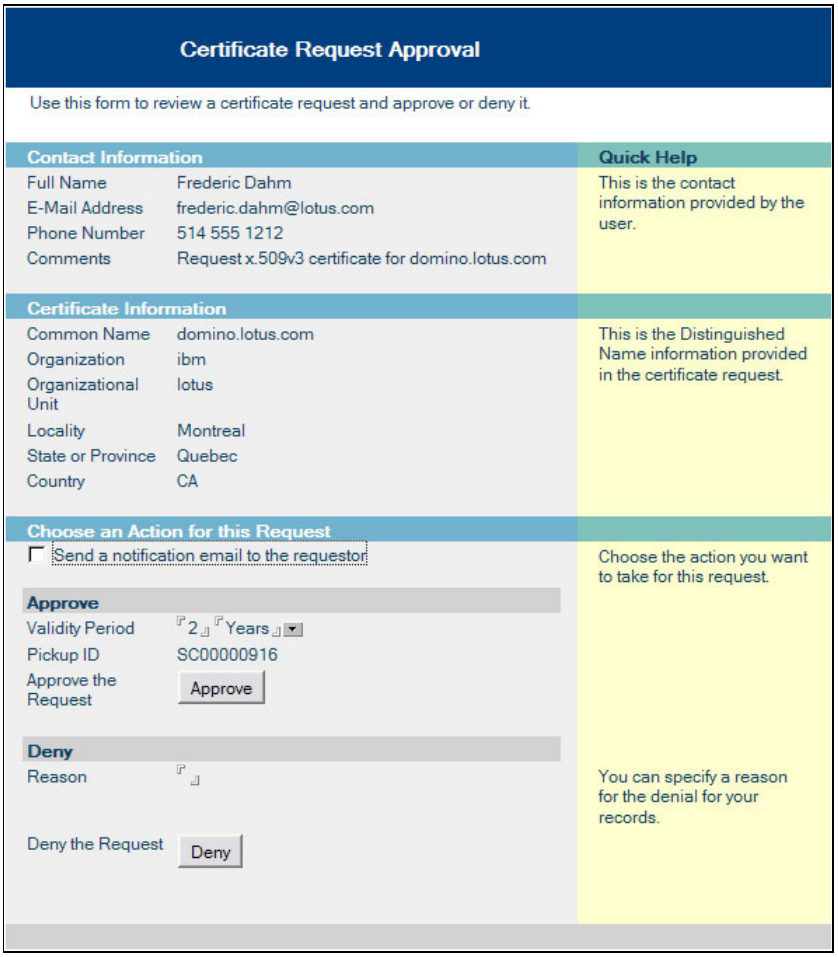


Figure C-20 The Certificate Request Approval form

c. There are only three fields on this form that might require some data entry:

- Send a notification e-mail to the requestor (check box)

  If you select this option or it is selected (it is selected by default if this was specified in the CA profile defaults), a key to access the certificate, called a “Pick Up ID” key (typically 10 alphanumeric characters), is sent to the requestor.

- Validity Period: **2 Years**

  This is the period (beginning at 0:00 UT on the day of approval) of validity of the certificate in years or days. You can change this default (again, derived from the CA profile) if you want.

- Reason: <blank, unless denied>

  This is where the administrator provides a reason if the request is denied.

If it is ascertained that the request is valid and can be approved, click **Approve** in the Approve section. Otherwise, click **Deny** in the Deny section. If there is any doubt, for the purposes of this exercise, click **Approve**. This opens the CA key ring password prompt dialog box, as shown in Figure C-21 on page 181.

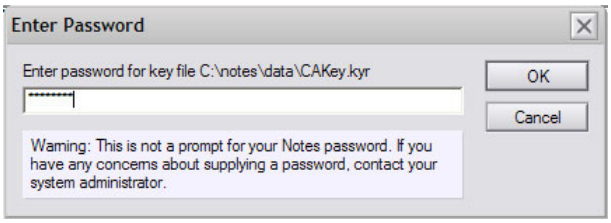


Figure C-21 The CA Key Ring Password Prompt dialog box

d. Enter the certificate authority's key ring file password so that it can be opened to retrieve the CA's private key to sign the certificate request. The request will be approved and will be placed in the view shown in Figure C-22 selectable through "View approved Certificate requests" (button with a green check mark).

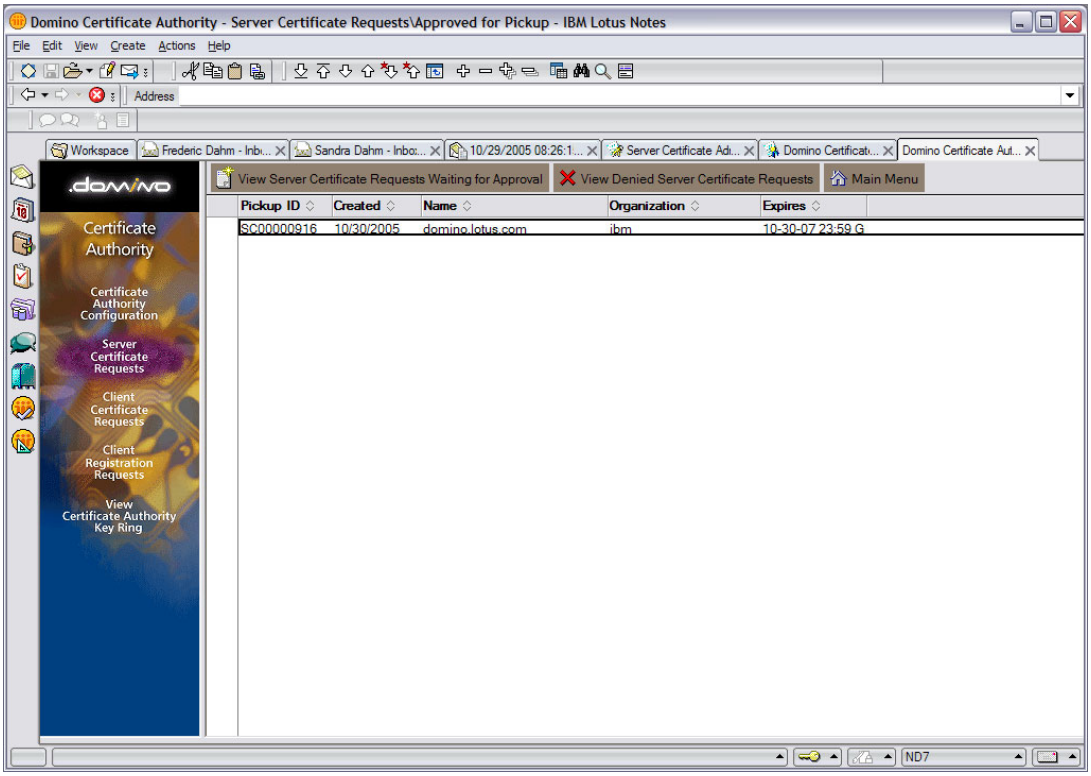


Figure C-22 The Server Certificate Requests\Approved for Pickup view

This completes the certificate authority's administrator's actions to approve the server certificate; it is now ready for the server administrator to pick up the certificate using the Pick Up ID.

2. To install the server's certificate into the server's key ring, perform the following steps:

   a. Using a Web browser, open the Domino Certificate Authority database (or the external CA you chose to certify your public key) and click **Pick Up Server Certificate** from the menu on the left. This opens the Pick Up Signed Certificate Web form on the right side of the page, as shown in Figure C-23 on page 182.

      Alternatively, if an e-mail notification was received saying that the certificate is approved and ready for pick up, it should have a URL that permits you to directly access the certificate ready for pick up. If this is the case, use the URL provided to open the Pick Up panel.

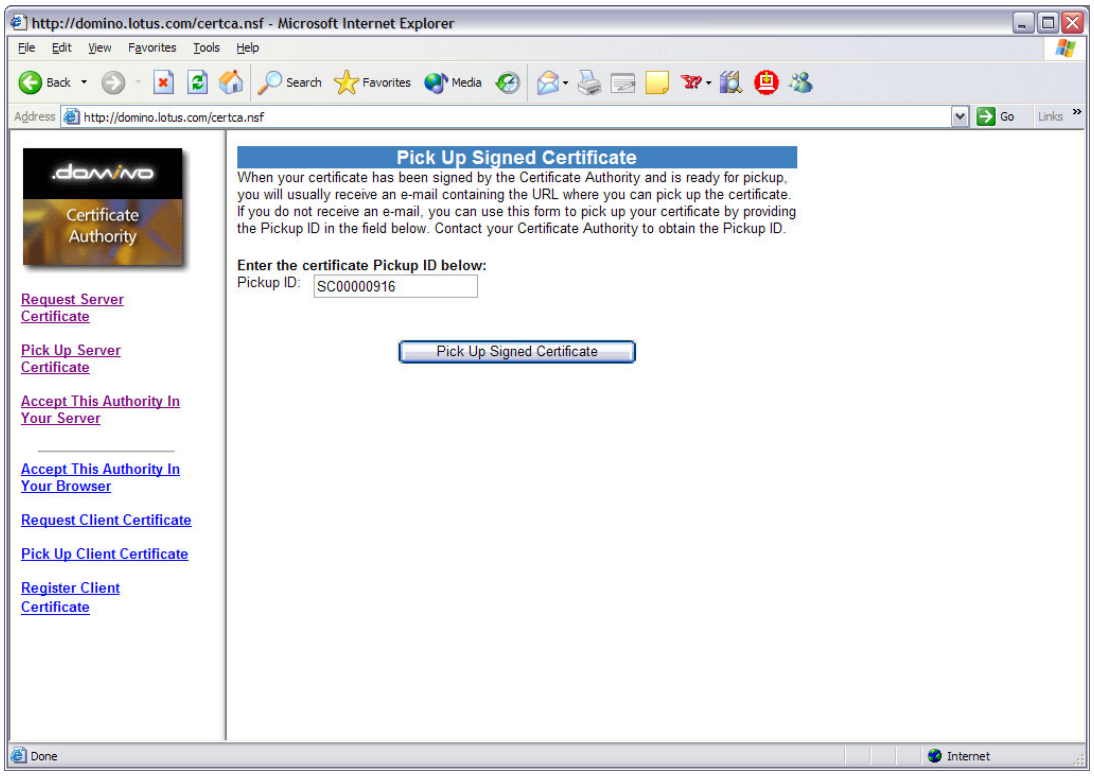


Figure C-23 The Pick Up Signed Certificate Web form

b. Enter the Pick Up ID in the field shown. (If you received notification by e-mail, it is most convenient to copy it to the Clipboard and paste it into the field; otherwise, type it into the field.) Click **Pick Up Signed Certificate**. This opens the Pick Up Signed Certificate confirmation document, as shown in Figure C-24 on page 183.

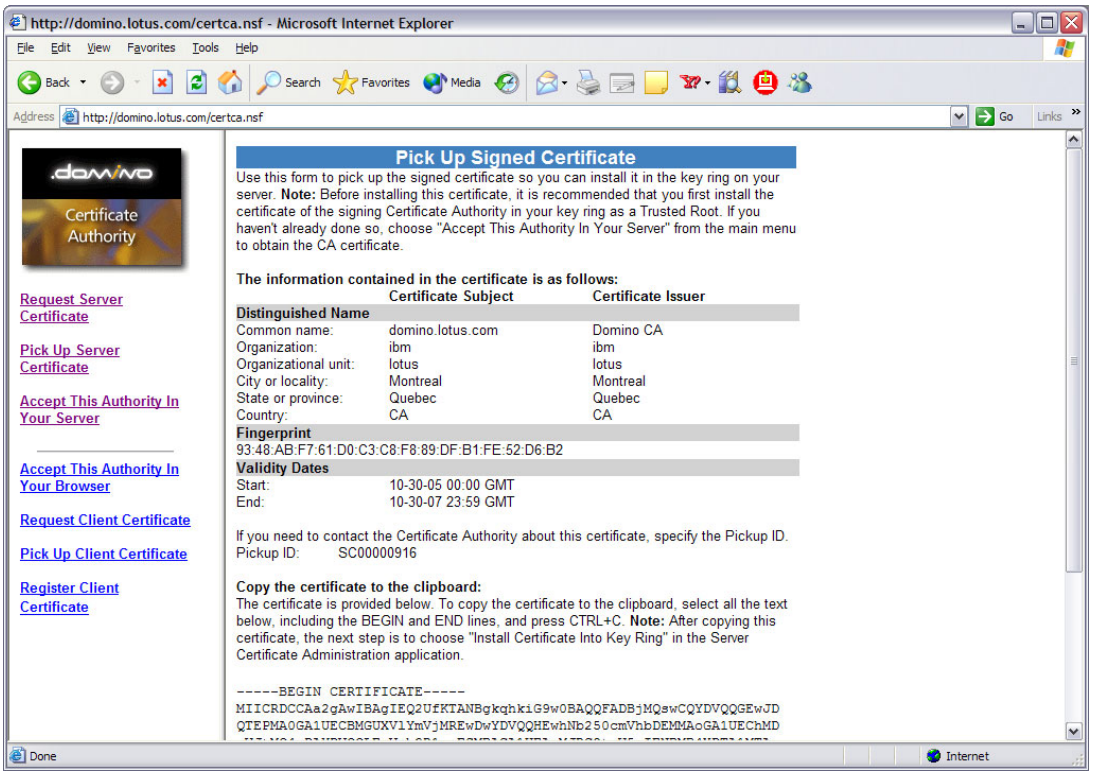


Figure C-24 The Pick Up Signed Certificate confirmation document

c. Scroll down to display the certificate, select it (be sure to include the “BEGIN CERTIFICATE” and “END CERTIFICATE” lines), and copy it to the Clipboard. Note that it is a good best practice to paste it to an instance of Notepad in case something requiring the clipboard is done between this step and the next step.

d. Using the Notes client, open the server certificate administration database (CERTSRV.NSF).

e. Select **4. Install Certificate into Key Ring**. This opens the Install Certificate into Key Ring form, as shown in Figure C-25 on page 184, which contains three fields that you should fill in, verify, and change if necessary:

- Key Ring File Name: `c:\notes\data\keyfile.kyr`

  This is the server key ring file name and path. This will have the value last typed into the field (for example, when you originally created the request).

- Certificate Source: **Clipboard**

  Select the certificate source (file or clipboard). The default is set to Clipboard.

- Certificate from Clipboard

  This is the certificate in PKSC-12 format.

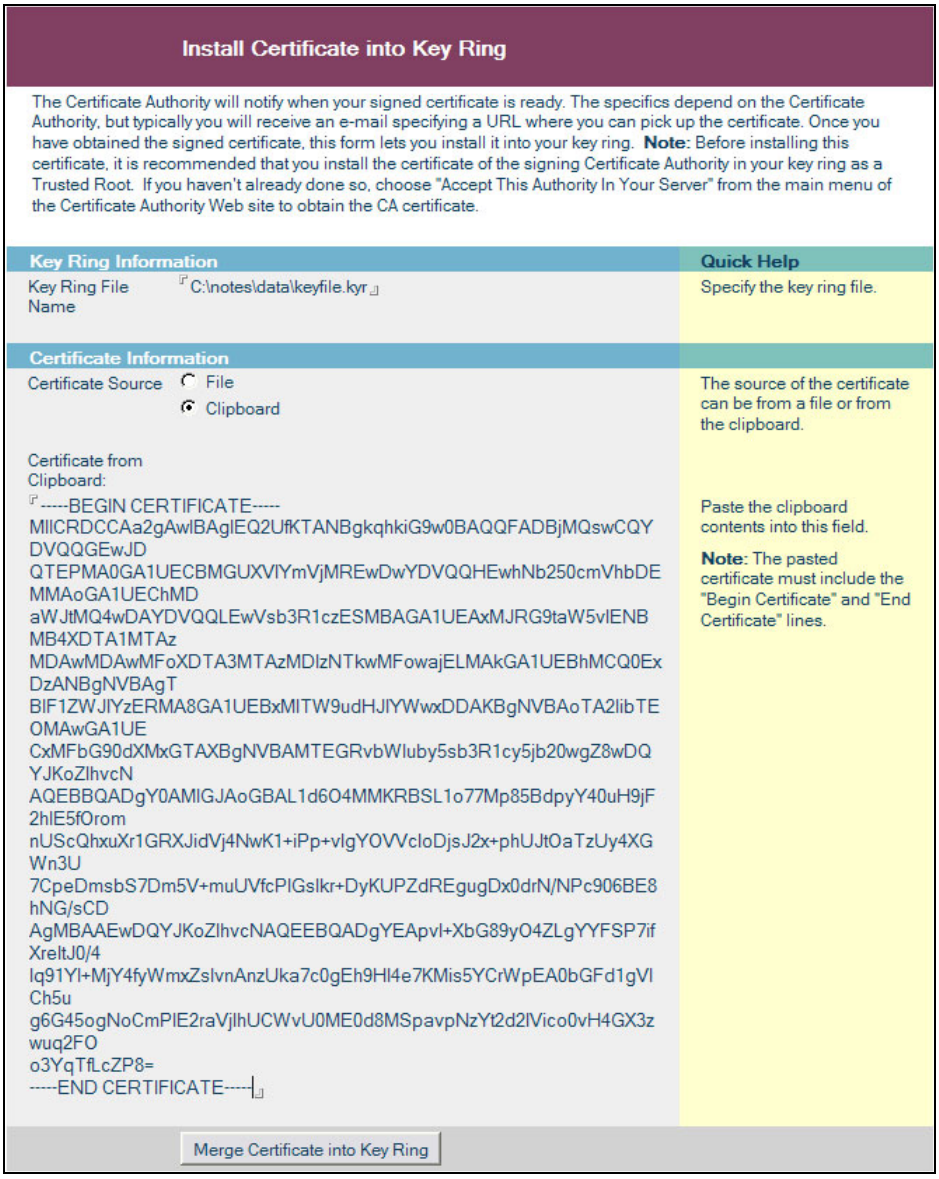


Figure C-25 The Install Certificate into Key Ring form

f. Click **Merge Certificate into Key Ring**. This opens the CA key ring password prompt dialog box, as shown in Figure C-26.

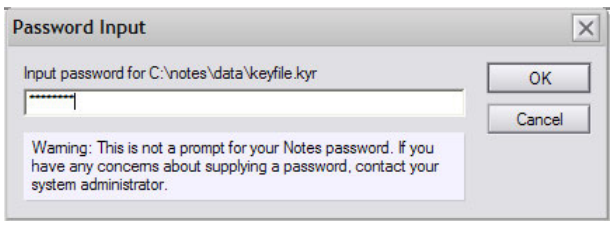


Figure C-26 The Server's Key Ring Password Prompt dialog box

g. When prompted, enter the password for the server's key ring file and click **OK**. A confirmation dialog box opens, as shown in Figure C-27 on page 185.

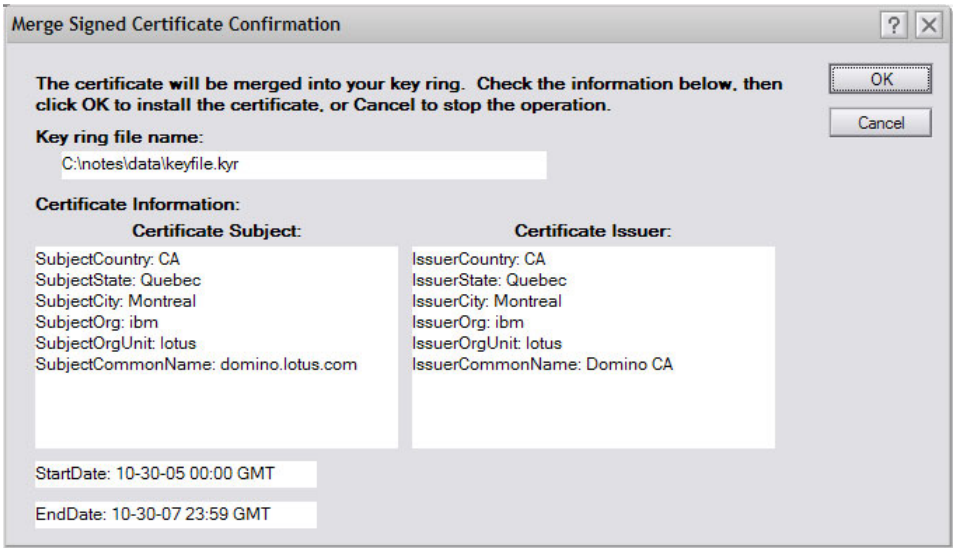


Figure C-27 The Merge Signed Certificate Confirmation dialog box

h. Click **OK**. The Certificate received into key ring message box opens, as shown in Figure C-28.

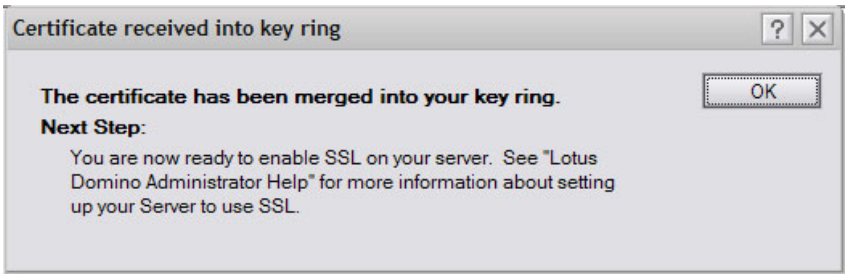


Figure C-28 The Certificate Received into Key Ring message box

i. Click **OK** to close the message box after the certificate has been merged into the key ring.

At this point, you can copy the key ring file onto the server's data disk (for example, c:\domino\data). Note that you must also copy the corresponding stash file (it will have the same file name as the key ring file, but with an extension of .sth). The stash file has an (encoded) copy of the server's key ring password so that the server can open the key ring.

In the next step, we reconfigure the server's HTTP task to use SSL by updating the Server document in the Domino Directory.

Step 6: Enable SSL on the Domino server

If this has not been done, move the server's key ring (keyfile.kyr) into the data directory of the Domino server before starting. Then, perform the following steps:

1. Open the Server document for the Domino server and ensure that it is in edit mode, as shown in Figure C-29 on page 186.

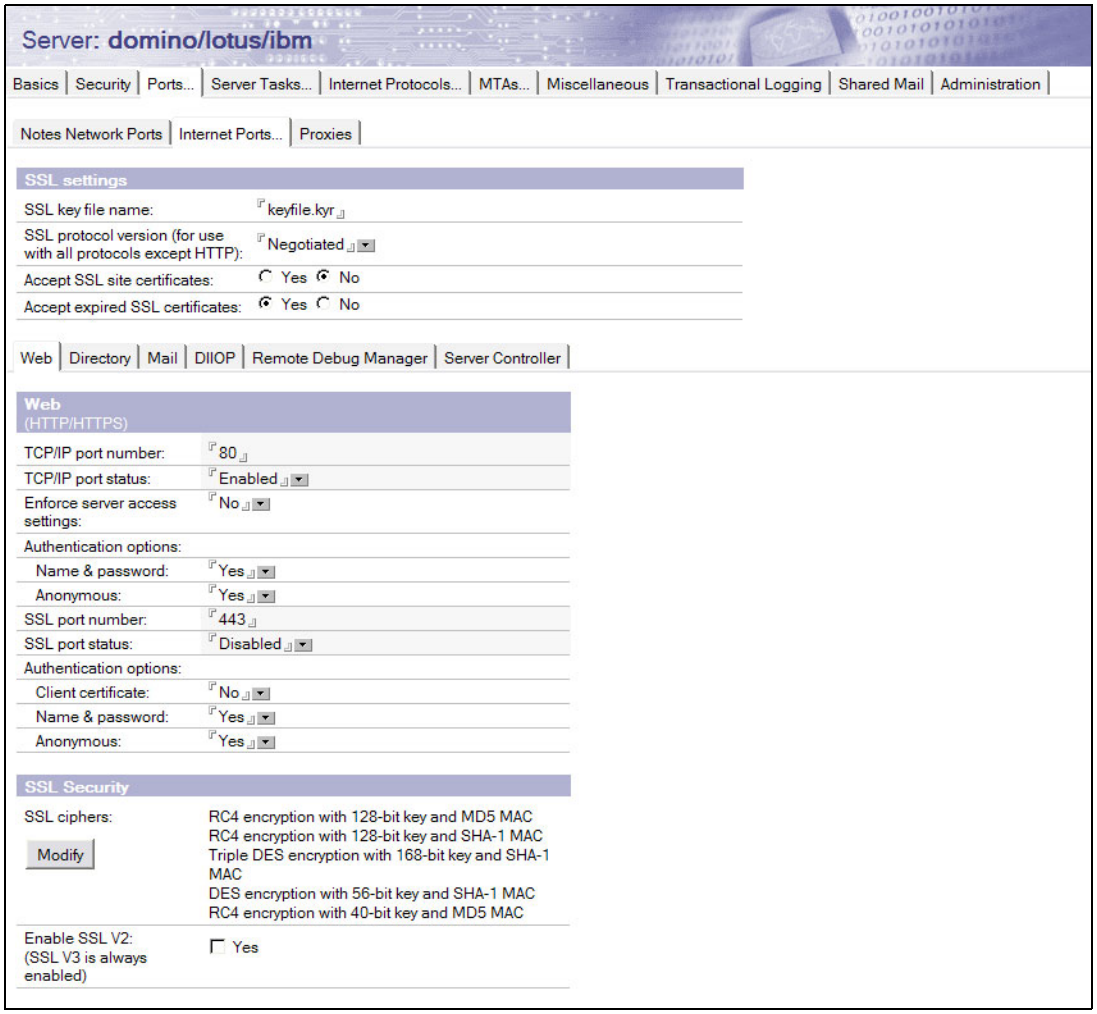


Figure C-29 The Domino Server Document: Web form

2. Go to the Internet Ports subtab and enter the following information in the SSL settings section (primarily ensuring that the proper key ring name is in the field “SSL key file name”):
   - SSL key file name: `keyfile.kyr`
   - SSL protocol version: **Negotiated**
   - Accept SSL site certificates: **No**
   - Accept expired SSL certificates: **Yes**
3. Go into the Web (HTTP/HTTPS) section, and change or enter the following information:
   - SSL port status: **Enabled**
   - Authentication options:
     - Client certificate: **No** (for SSLv3 browser certificates)
     - Name & password: **Yes** (for basic name and password authentication)
     - Anonymous: **Yes** (for anonymous access)

Note: At this time, because we have not issued any client certificates, Client certificate authentication should be set to **No**; otherwise, it will cause confusion for users when they are prompted to perform SSL authentication and have no certificates to present the Domino server.

4. Click **Save and Close**. Note that if you edited the Server document on a different server from the one in the Server document, replicate the NAMES.NSF database to the remote server before trying to start SSL.

SSL will restart the moment the Domino is restarted, or more specifically when the HTTP task is restarted. Use the **`tell http restart`** command to restart the HTTP task to immediately enable SSL (although you can also issue the **`tell http quit`** command followed by the **`load http`** command).

You have now installed and enabled SSL on the server.

Step 7: Test SSL

There are two ways of testing to see if the configuration of SSL is correct and that SSL is in operation.

The first way is to go to the server's console and enter the following command:

```
tell http show security
```

The server should return the following information:

```
10/30/2005 07:28:18 PM Base server:
10/30/2005 07:28:18 PM SSL enabled
10/30/2005 07:28:18 PM Key file name: c:\domino\data\keyfile.kyr
10/30/2005 07:28:18 PM Secure server started
```

The second way to test that SSL is enabled is by accessing your server using a URL of the following form:

```
https://domino.lotus.com/names.nsf?Open
```

If the database can be accessed with the same result as though it had been opened without SSL (with the difference that with SSL, the lock is locked in the Web browser), the test is successful.

It is likely that security alerts will be triggered by testing SSL at this stage. If the Domino CA was used as the CA for this exercise, it is not listed as a trusted root in the Web browser's certificate store. Therefore, the security certificate presented by the server cannot be trusted, which triggers the warning, as shown in Figure C-30 on page 188.

Figure C-30 Security Alert: Untrusted certificate

If you proceed after the security alert (by clicking **Yes**), another security warning opens, as shown in Figure C-31. If you proceed (again, by clicking **Yes**), the URL will be loaded in the same manner as though there had not been any security warnings.

Figure C-31 Further Security warning after decision to proceed

If client certificate authentication has been enabled, this generates an erroneous condition, and the Client Authentication dialog box opens, as shown in Figure C-32 on page 189. It is empty because, at this stage, a client certificate was neither generated for the user nor merged into the Web browser's certificate store.

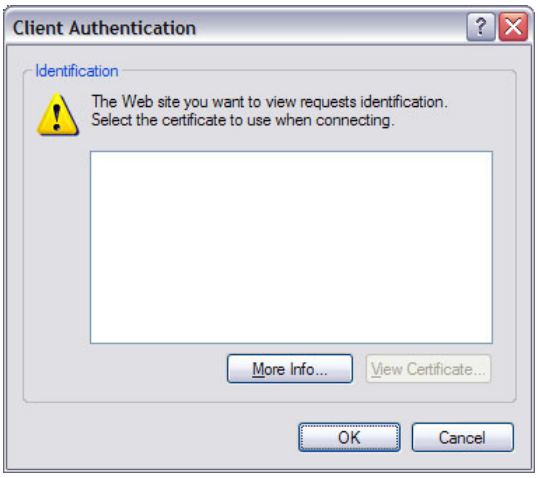


Figure C-32 Client Authentication problem

Step 8 (Optional): Accept the CA as a trusted root in the Web browser

This is very similar to the server process described in “Step 4: Merge the certificate authority’s certificate into the server key ring” on page 175 except that the CA trusted root certificate will be added to Web browser's key ring.

By accepting a certificate authority as a trusted root in your Web browser, you are setting up a trust relationship: You (or rather your browser) will accept any certificates signed by this CA as valid because you have indicated that you trust the CA.

Note that if you request a certificate from an external CA, you might already have its certificate installed by default. To check using Microsoft Internet Explorer (we used Internet Explorer 6.0 for this test; your version might be slightly different), perform the following steps:

1. Select **Tools** → **Internet Options** from the Windows menu.
2. Go to the Content tab and click **Certificates**.
3. Go to the Trusted Root Certification Authorities tab. This lists the default certification authorities present in the Web browser, as shown in Figure C-33 on page 190.
4. From this dialog box, you can remove the flag “trusted root” for any certificate or delete it from your key ring.

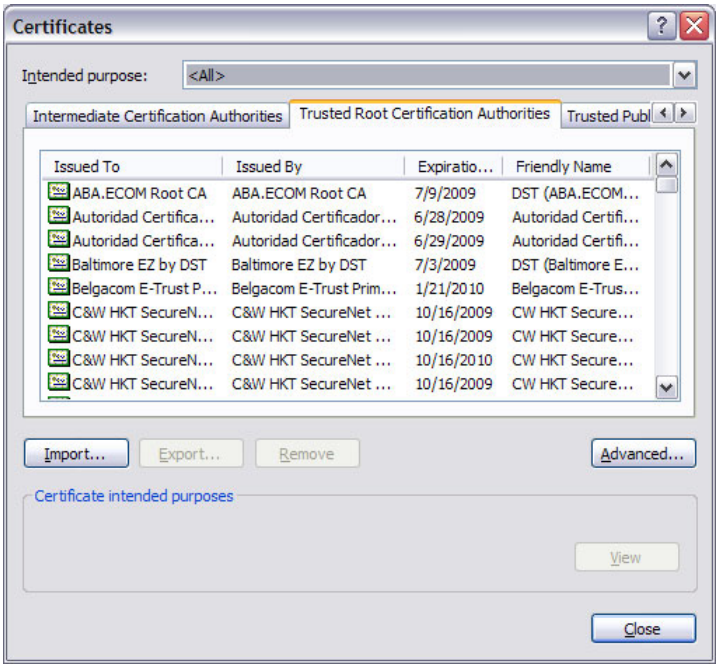


Figure C-33 The Trusted Root Certification Authorities present in the Web browser

To accept the server's certificate in the Web browser, perform the following steps:

1. Using the Web browser, open the Domino Certificate Authority database (CERTCA.NSF) and select **Accept This Authority In Your Browser**. This opens the Trust This Authority in Your Browser document, as shown in Figure C-34.

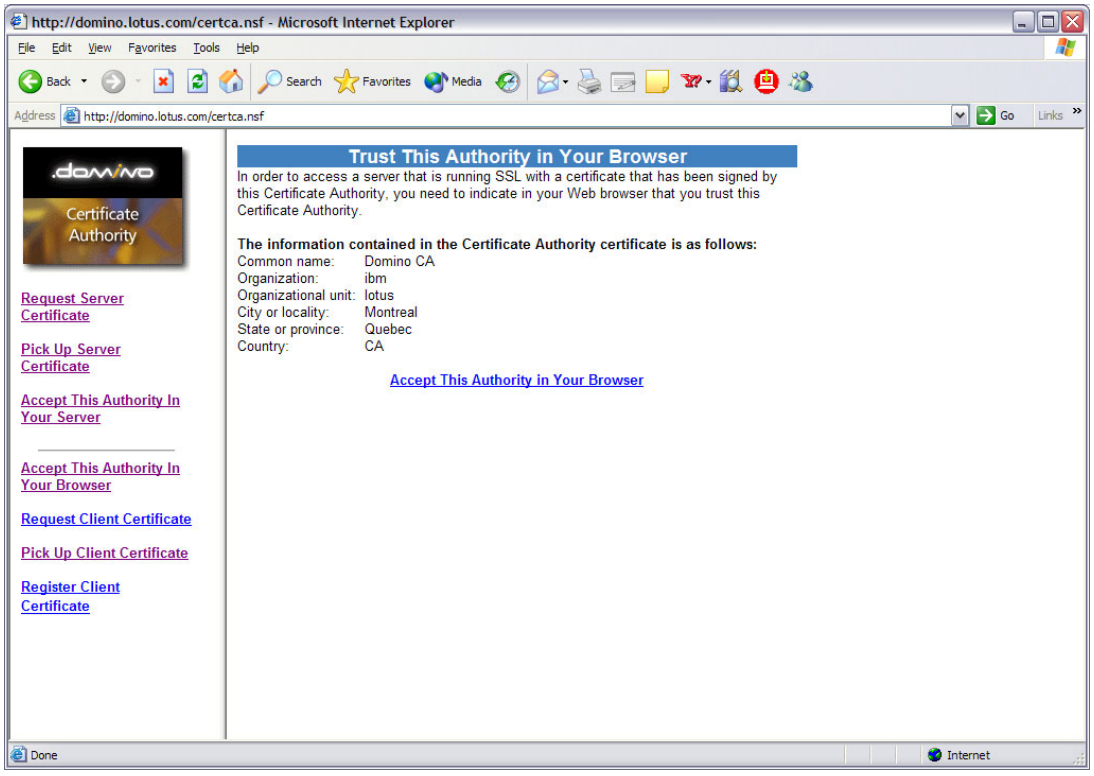


Figure C-34 The Trust This Authority in Your Browser document

2. Click the **Accept This Authority in Your Browser** link on the page. This opens the File Download dialog box, as shown in Figure C-35.

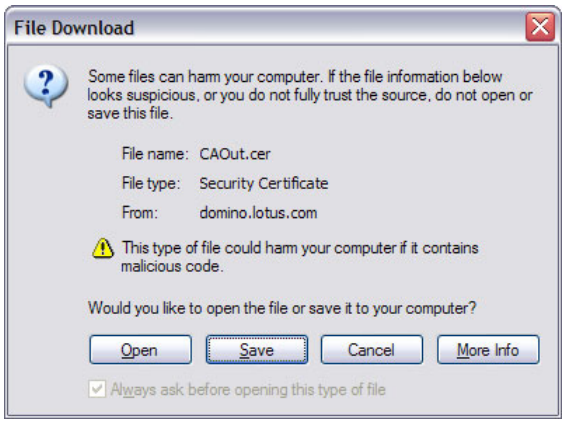


Figure C-35 The File Download dialog box

3. Click **Open**. After the Security Certificate downloads from the server, the Certificate information dialog box opens, as shown in Figure C-36.

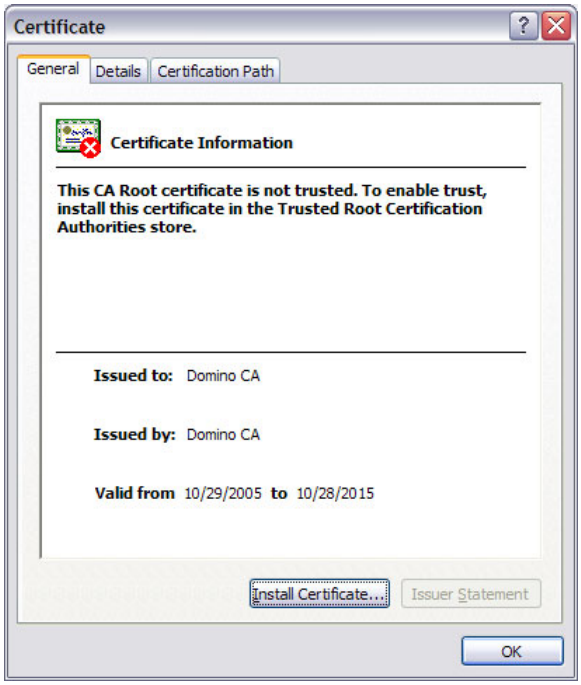


Figure C-36 The Certificate information dialog box

4. This is where it might be somewhat confusing. If you click **OK**, the Certificate information dialog box closes and the certificate will not be accepted in the Web browser (although you might think it was accepted). Therefore, in the Certificate information dialog box, click **Install Certificate**. The Welcome to the Certificate Import Wizard dialog box opens, as shown in Figure C-37 on page 192.

Figure C-37 The Welcome to the Certificate Import Wizard dialog box

5. Click the **Next**.The next pane of the Certificate Import Wizard opens, as shown in Figure C-38.

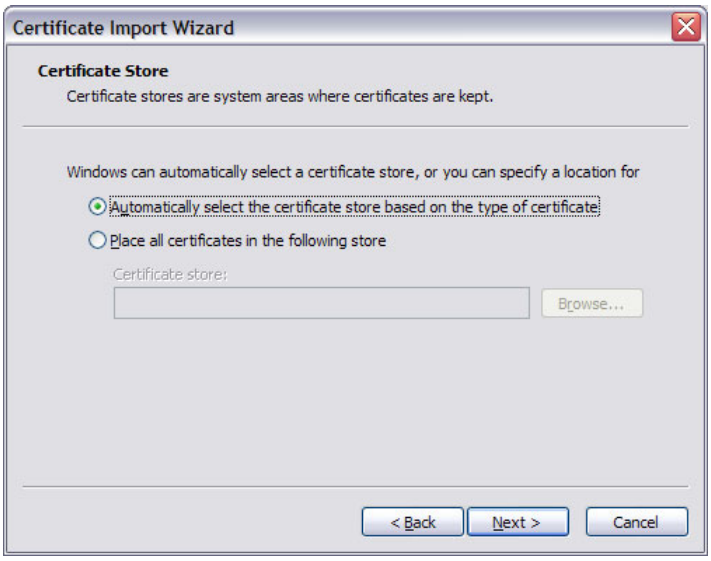


Figure C-38 The next pane of the Certificate Import Wizard

6. Ensure that the **Automatically select the certificate store based on the type of certificate** option is selected and click **Next**. The final pane of the Certificate Import Wizard opens, as shown in Figure C-39 on page 193.

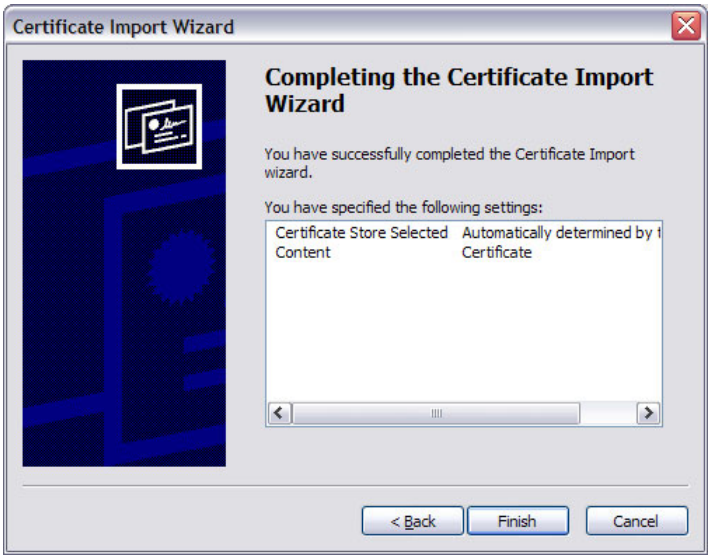


Figure C-39 The final pane of the Certificate Import Wizard

7. Click **Finish**. The Root Certificate Store addition query dialog box opens, as shown in Figure C-40.

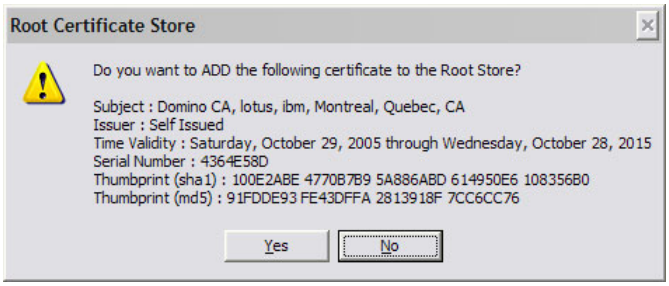


Figure C-40 Query dialog box for adding a certificate to the root store

8. Click **Yes** to add the certificate to the Web browser's Web store. A final message box opens confirming that the import was successful, as shown in Figure C-41.

Figure C-41 Confirmation of the import

Now, it is possible to check again with Microsoft Internet Explorer (we used Internet Explorer 6.0 for this test; your version might be slightly different). Perform the following steps:

1. Select **Tools** → **Internet Options** from the window menu.
2. Go to the Content tab and click **Certificates**.
3. Go to the Trusted Root Certification Authorities tab. This lists the default certification authorities in the Web browser, as shown in Figure C-42 on page 194.
4. From this dialog box, check to see if the CA has been added as a trusted root. Scroll down the list, and you should see the Domino CA trusted root.

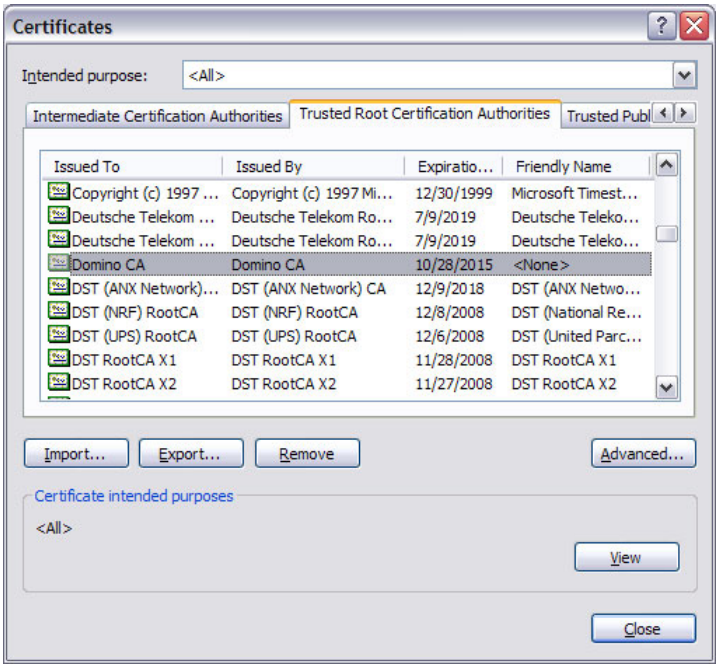


Figure C-42 Checking the Trusted Root Certification Authorities

Your tasks are now complete, unless you need to do client certificate authentication or use secure messaging using S/MIME. In this case, you must generate, pick up, and use client certificates. We describe this in the following section.

Requesting, picking up, and using a client certificate

In this section, we describe the steps involved in requesting, picking up, and using a client certificate. This type of certificate permits us to do SSLv3 client authentication and provides a certificate to use secure messaging using S/MIME.

Step 1: Request a client certificate

Perform the following steps to request a client certificate:

1. With a Web browser, open the Certificate Authority database (CERTCA.NSF). Select **Request Client Certificate**.
2. This opens the “Request a Client Certificate for Microsoft Internet Explorer” form, as shown in Figure C-43 on page 195. Enter the following information.

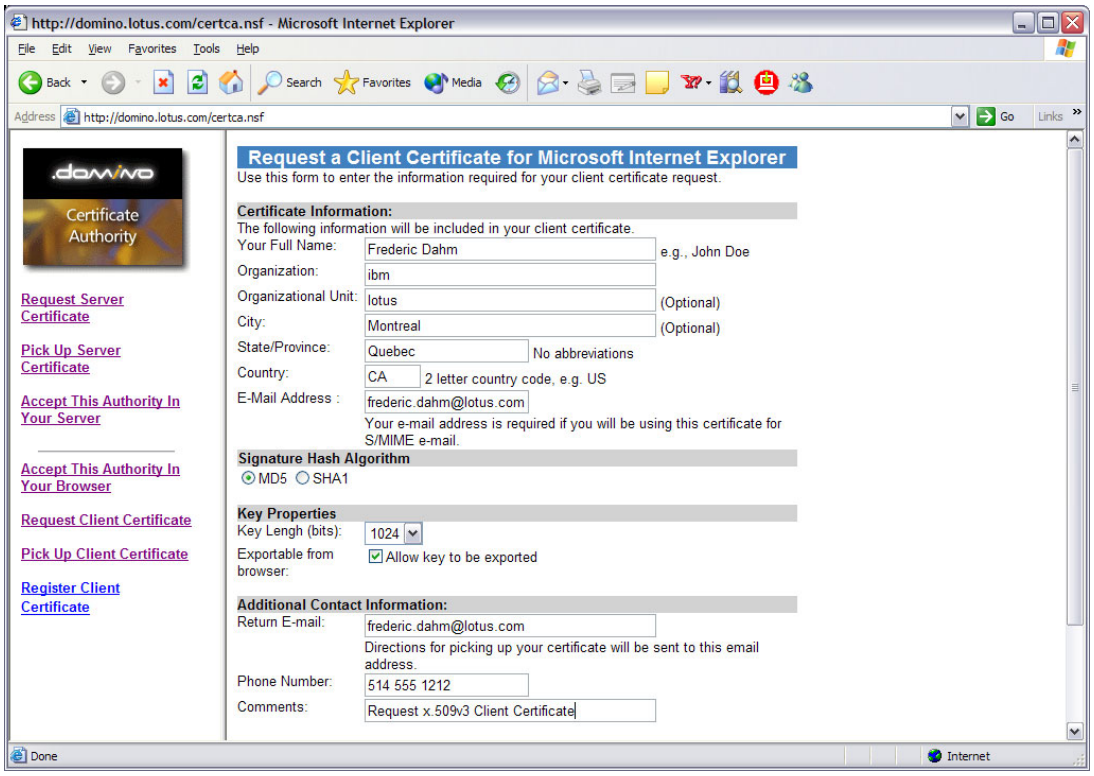


Figure C-43 The Request a Client Certificate for Microsoft Internet Explorer form

3. Click the **Submit Certificate Request** link at the bottom of the form. This might open Potential Scripting Violation message box, as shown in Figure C-44.

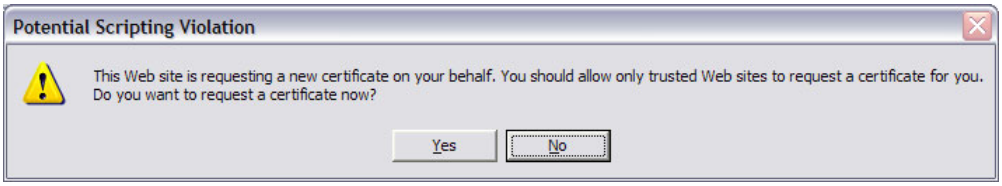


Figure C-44 The Potential Scripting Violation message box

4. Click **Yes** in the Potential Scripting Violation dialog box. A response from the CA confirming receipt of the certificate request opens, as shown in Figure C-45 on page 196.

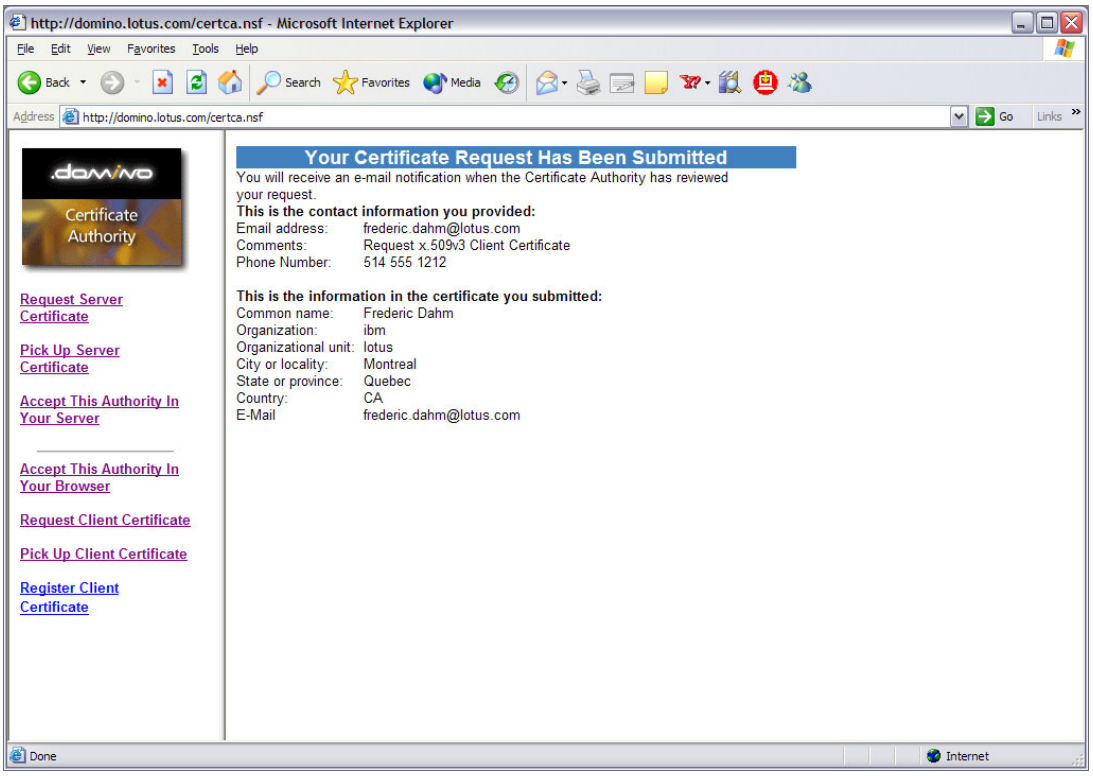


Figure C-45 The Your Certificate Request Has Been Submitted document

This completes the process of requesting a certificate. The certificate authority has to approve your request before you can proceed.

Step 2: Approve a client certificate request in the Domino CA

Perform the following steps:

1. Using the Notes client, open the Certificate Authority database and select **Client Certificate Requests** from the menu on the left side of the page. The Client Certificate Requests\Waiting for Approval view opens, as shown in Figure C-46 on page 197.

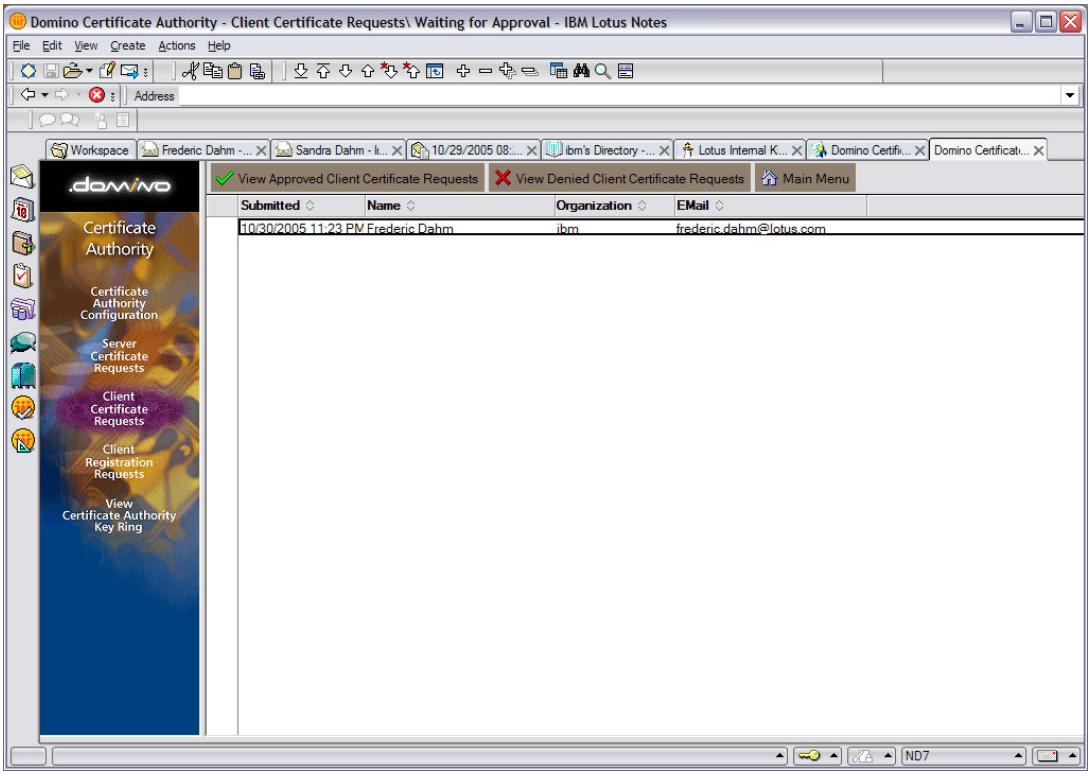


Figure C-46 The Client Certificate Requests\Waiting for Approval view

2. Select the client request or requests to be approved (there is only one in our example). Open the request to approve it. The Client Certificate Request Approval form opens, as shown in Figure C-47 on page 198.

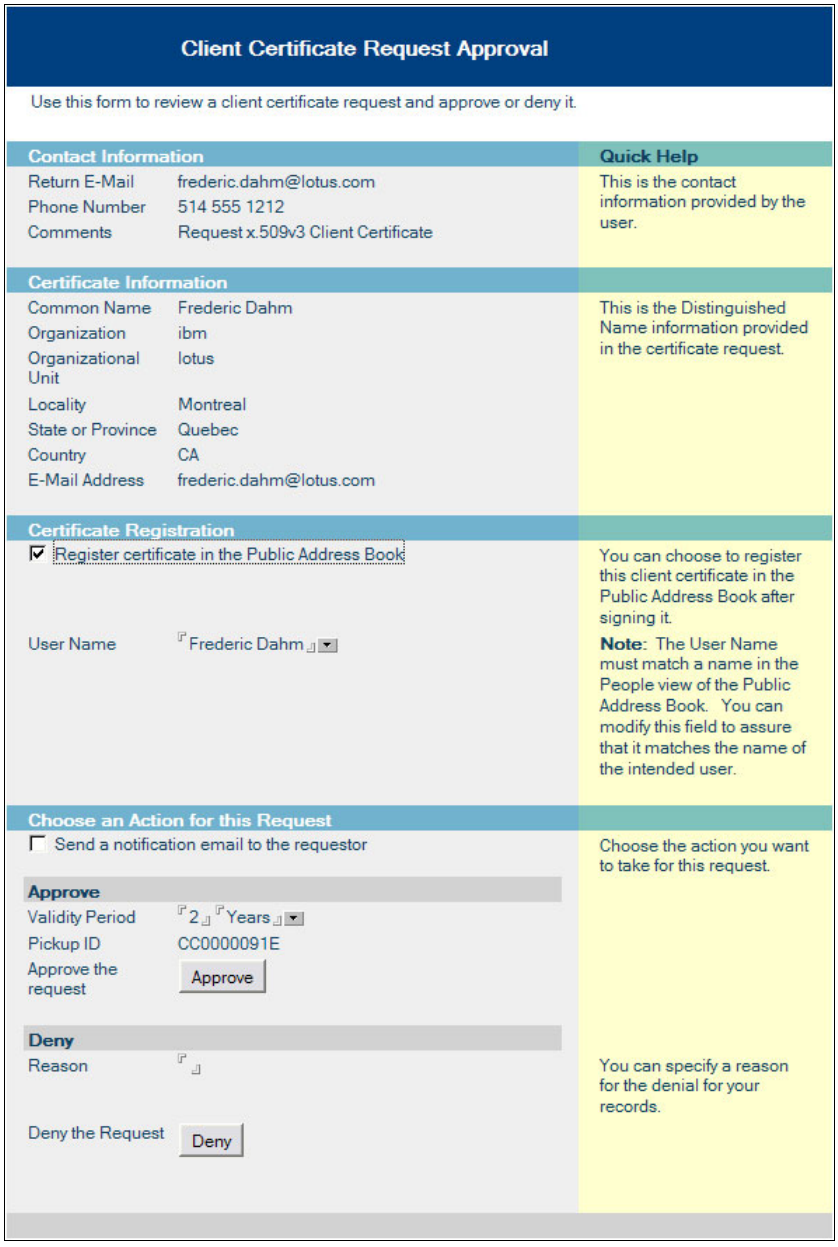


Figure C-47 The Client Certificate Request Approval form

3. The top of the form has the information from the client certificate request (Contact Information and Certificate Information sections). Note that the contact information does not need to be the same as the certificate information. They can be different if the submitter chose a colleague as a contact because, for example, a newly hired person without a telephone wanted to register their certificate. Note also that you can only view the certificate information. If it is incorrect, you need to deny the request and ask the client to resubmit a corrected request.

– Certificate Registration:

  • Register certificate in the Public Address Book

    This option enables you to have the certificate, when approved, to be added to the requestor's Person document (Internet Certificates subtab of the Certificates tab). If selected, a request is placed in the Administration Requests database to perform the action. It is performed on the administration server for the Domino Directory; this might not be the same as the server used for the Domino certificate authority. This requires that you ensure that the person is already in the Domino Directory (preferably on all replicas, but it must be in the replica on the administration server for the Domino Directory when the request is performed).

    If you do not select this action at this time, you can still do so in the future by opening the approved client request in the Certificate Authority database. There will be an information section telling you that you did not add this person's certificate to the Domino Directory. You will have a User name field to look up the requestor's name in the Directory and a button to submit the request.

  • User Name: **Frederic Dahm**

    The name to locate the person in the Directory can be changed from this field. This enables the administrator to change it to match one of the entries in the User name field of the user's Person document. The submitted entry can be different if the certificate had a slightly different spelling or format, but this should not invalidate adding it to the Domino Directory. You can select the down arrow to the right of the field to open the Domino Directory (any directories known to your client) to search for names and select the desired name.

– Choose an Action for this Request:

  • Send a notification email to the requestor

    This option enables you to select that an automatic e-mail will be sent to the requestor. It is selected by default if you selected this option in the CA profile. Note that users should be cautioned that, even when their request is approved, there might be a delay before their certificate is added to the Domino Directory and, when added, propagated to all replicas of it. The default message in the generated e-mail implies that the process is immediate.

  • Approve:

    Validity Period: **2 Years**

    Again, this will be set to 2 years by default, but you can change it to any period (in years or days) that you want.

    Pickup ID: <Pickup ID> (for example, CC0000091E)

  • Deny:

    Reason: <>

    If, for some reason, the administrator denies the request, the administrator can fill out this field. It is saved with the request for future reference and included in the e-mail to the requestor.

4. If it is ascertained that the request is valid and can be approved, click **Approve** in the Approve section. Otherwise, click **Deny** in the Deny section (if there is any doubt, for the purposes of this exercise, click Approve). This opens the CA key ring password prompt dialog box, as shown in Figure C-48 on page 200.

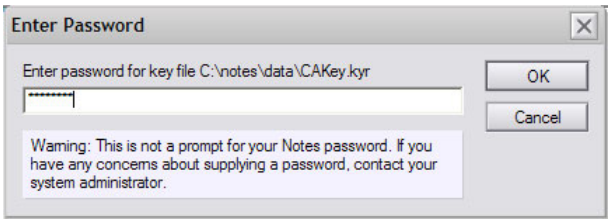


Figure C-48 The server's key ring password prompt dialog box

If you selected to add the certificate to the Domino Directory by the Domino Administration Process, you can confirm that the addition was successful after a reasonable delay for processing and replication. Open the user's Person document, and go to the Certificates tab and the Internet Certificates subtab. Inspect their Internet certificate, as displayed in Figure C-49. If present, the addition was successful. No further action is required other than ensuring that the Domino Directory replicates to all servers with the LDAP task enabled and pointed to by the Directory Assistance database for Web browser authentication.

If the LDAP task is running, You can also search for the certificate by running the **`ldapsearch`** command included with the Domino server. Run the following command from the command line:

```
ldapsearch -v -L -h domino.lotus.com "cn=Fred*"
```

The presence of (binary) data in the "user certificate" field confirms the presence of a certificate in Frederic Dahm's Person record.

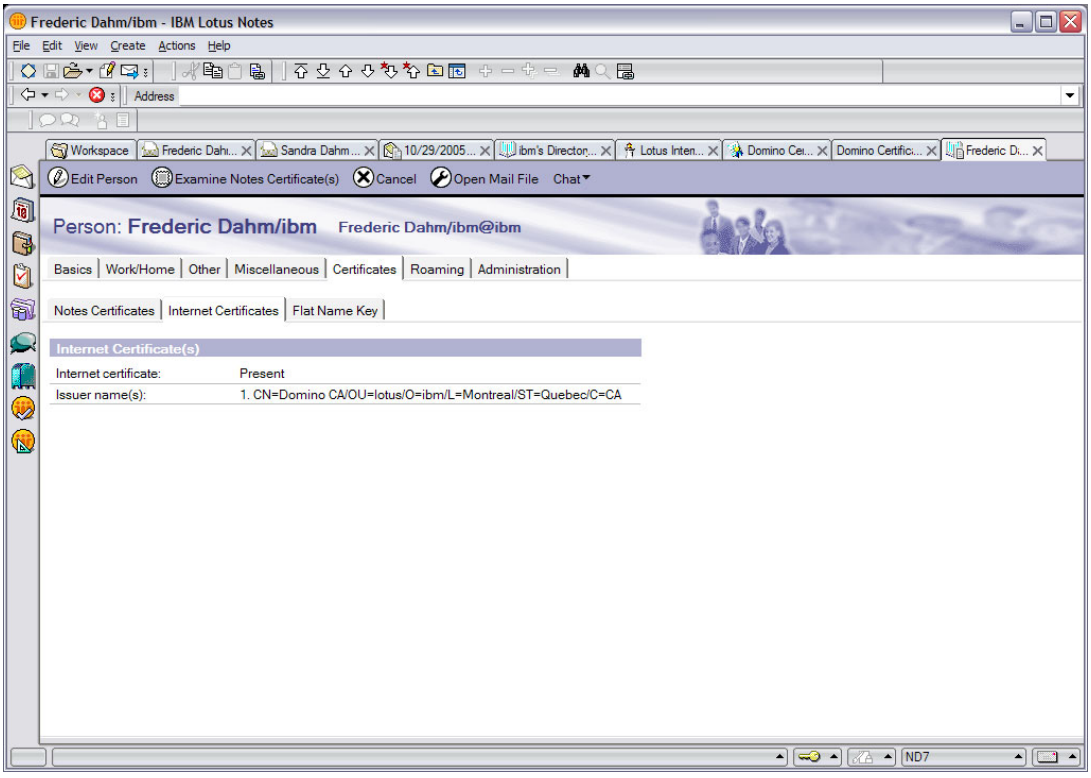


Figure C-49 The Person document in the Domino Directory (Internet Certificates)

If the certificate is not (yet) present in the Person record in the Domino Directory, you can:

1. Open the Administration Requests database.

2. Open **All Requests by Server** from the navigator on the left.
3. Expand the category **Administration Server of Public Address Book** and the Add **Internet Certificate to Person Record** subcategory. You should find your request (you might have to scroll to it if there are many requests) and the action performed on it. There will be a response document under the original request if it has been processed.

   Note that if the processing was successful, there will be a green check mark showing in the view next to the response document, as shown in Figure C-50. No further action is required (other than ensuring that the Domino Directory replicates to all servers). Note also that if there is a red X next to the response document, the addition failed. Open it to discover why the request failed. Correct the situation and resubmit the request from the response document.

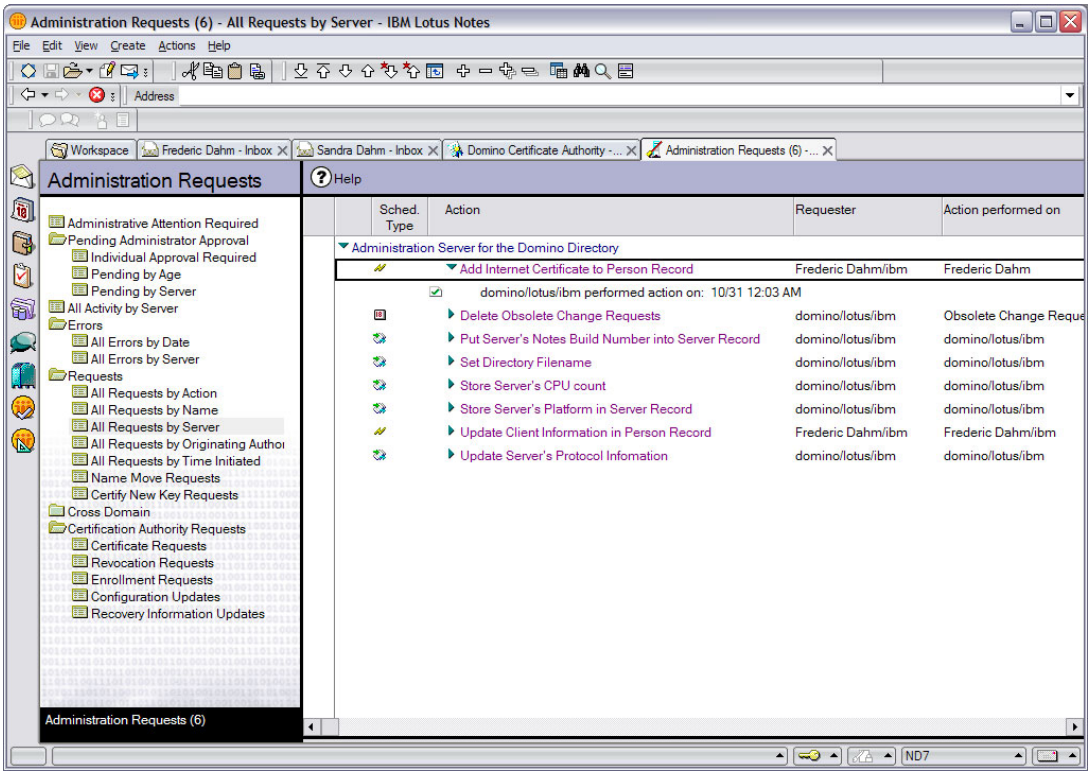


Figure C-50 The administration request for adding an Internet certificate to the Person record

This completes the approval process. The client certificate is now ready to be picked up by the client browser.

Step 3: Accept a client certificate into a browser key ring

A confirmation e-mail can be sent from the certificate administrator telling the user that the certificate has been approved and is ready for pick up. If so, it might have a URL that can be pasted by the user into the Web browser to directly access the certificate ready for pickup.

It is also possible for the user to manually enter the correct URL in the Web browser to point to the Domino CA and select **Pick Up Client Certificate**. The user will be prompted for the Pickup ID (this is part of the URL in the e-mail), which will have been sent to the user by the certificate administrator. The panel is very similar to the one to pick up a server certificate.

Use the following steps:

1. Using a Web browser, open the Certificate Authority database (CERTCA.NSF) and select **Pick Up Signed Certificate**. The Pick Up Signed Certificate form opens, as shown in Figure C-51.

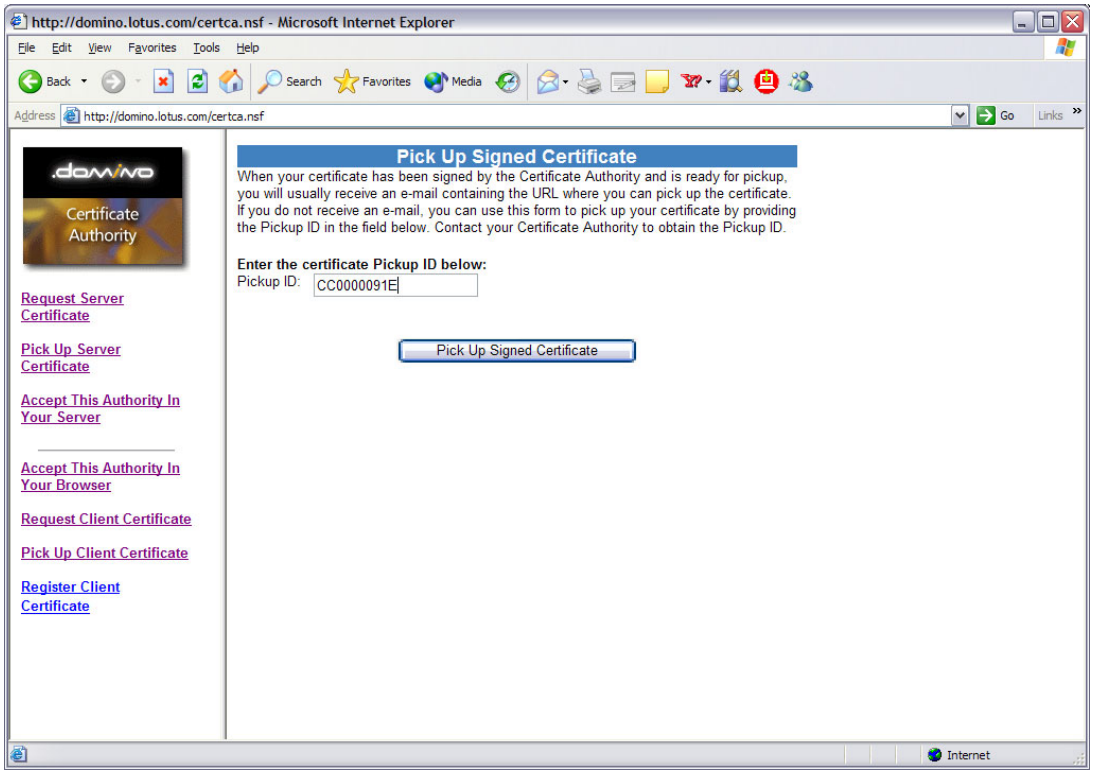


Figure C-51 The Pick Up Signed Certificate form

2. In the Pick Up Signed Certificate, paste (or enter) the pickup ID (for example, `CC0000091E`) and click **Pick Up Signed Certificate**. The Pick Up Signed Client Certificate for Microsoft Internet Explorer document opens, as shown in Figure C-52 on page 203.

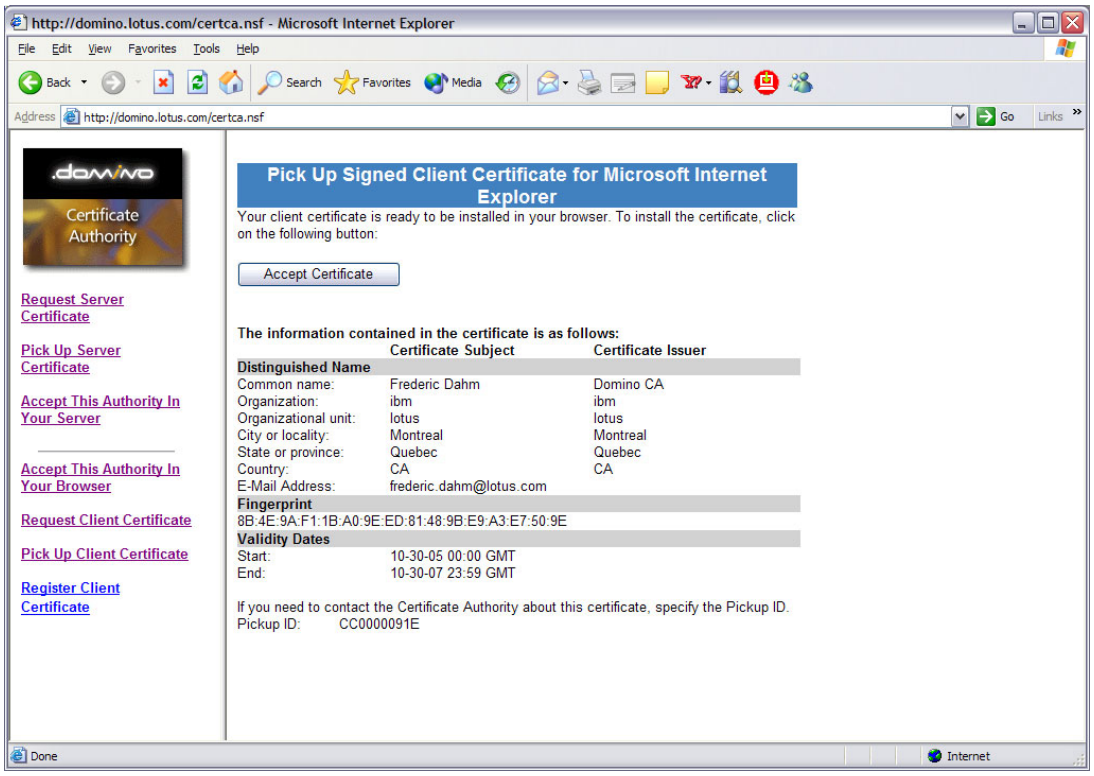


Figure C-52 The Pick Up Signed Client Certificate for Microsoft Internet Explorer document

3. Click **Accept Certificate** after confirming that the certificate is valid. The client certificate will be installed into your browser's key ring. Note that there is not a confirmation message. A Potential Scripting Violation message box opens, as shown in Figure C-53.

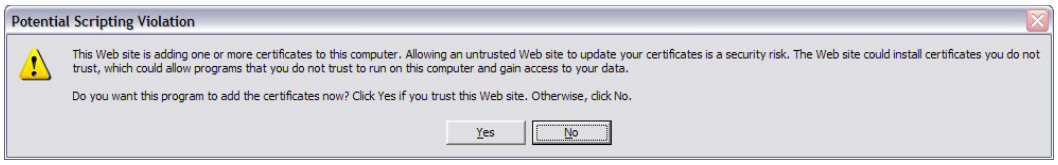


Figure C-53 The Potential Scripting Violation message box

4. Click **Yes** in the Potential Scripting Violation dialog box. A response from the CA confirming receipt of the certificate request opens, as shown in Figure C-54.

Figure C-54 The Certificate Successfully Installed message box

5. Click **OK** in the Certificate Successfully Installed message box.

To check in Microsoft Internet Explorer to see if the client certificate is in one of the certificate store, select **Tools** → **Internet Options** from the Windows menu,. Go to the Content tab, click **Certificates**, and go to the Personal tab. To view the contents of the certificate, click the entry in the list and then click **View**. A dialog box listing the default certification authorities

present in the Web browser opens, as shown in Figure C-55. Click **OK** to close the certificate dialog box and **Close** to close the Certificates dialog box.

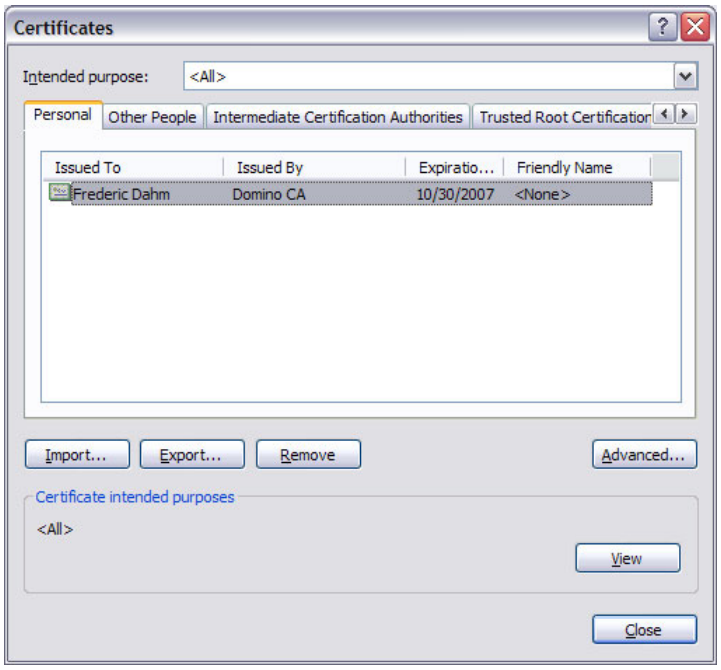


Figure C-55 Checking the Client Certificates

This completes the process of requesting and installing a client certificate in a Web browser.

If you used a Domino CA, your certificate administrator had the opportunity to add your public certificate to your Person record in the Domino Directory either while approving your request or subsequently by viewing the approved request and requesting registration. If so, no further action is necessary on your part. If you received your certificate from an external certificate authority or your administrator did not copy your certificate to the Domino Directory, you can now request that this be done.

Step 4: (Optional) Request registration of a client certificate

This process exists to enable a client with a Web browser certificate to request that a copy be placed in their Person record in the Domino Directory. This puts a request record in the Domino Certificate Authority database, which the administrator will need to approve.

If you have a certificate from the (same) Domino CA, you do not need to do this: If the administrator has a copy of your approved certificate request, the administrator can request that a copy be placed in your Person record by clicking a button on the approved request form. If the request is no longer present in the database or you received your certificate from another CA, you must follow this process to have your browser certificate copied the Domino Directory. This is necessary if your company uses certificate-based LDAP authentication. Your administrator must already have configured the server on which the Domino CA runs to support SSL as described in “Requesting and installing a server certificate” on page 167.

Perform the following steps:

1. To start the process, with a Web browser, navigate to the Domino Certificate Authority database (CERTCA.NSF).
2. Specify SSL (by using `https://` in the URL) and select **Register a Client Certificate**.

3. You will be asked to fill in a panel with your contact information. Note that this information is only for the administrator because your “official” information is contained in your certificate.
4. Click **Submit Certificate** when you have entered your contact information. If you receive an error message (for example, “Unable to process certificate registration request. The request requires a valid certificate signed by a recognized Certificate Authority, and must be submitted over an SSL connection.”), you need to resubmit the request.

   The error message “Unable to process certificate registration request” can result from one of two possibilities:

   – You did not specify SSL. Correct the URL to specify “`https:`” and retry the request. You might need to clear your browser’s cache or restart it. Ensure that you specify SSL when opening the Certificate Authority database, not just when you submit the request. This is because the client certificate is requested as part of the SSL handshake when you first open the database.

   – The server the Domino CA uses does not support SSL. Contact your administrator to correct this and retry when it supports SSL.
5. When you connect successfully using SSL, a dialog box opens requesting you to select which client certificate you want to use. If you have more than one certificate, select the one you want to have registered in the Domino Directory (expand the list by clicking the arrow to the right of the Select Your Certificate field). Inspect the certificate by clicking **More Info**. When you have selected the correct certificate, click **Continue**. You will receive a confirmation panel.

This completes your request. Your administrator must now approve it for the request to be processed. They have the option of e-mailing you a confirmation that the request has been approved. Note that the actual copy into the Domino Directory might not be immediate because it is performed by a background system process (the Administration Process) and might have to performed on a different server than the one to which your request was submitted. You will not have to perform any further action after the request is submitted.

For the administrator to approve the client registration request, perform the following steps:

1. Open the Domino Certificate Authority database and select **Client Registration Requests** from the menu on the left. This action opens a view of outstanding client certificate registration requests.
2. Select the document with the request to be approved and open it. An approval panel opens. The top of the form has the contact information from the registration request. The contact information does not need to be the same as the certificate subject information displayed in the “Certificate Subject” area at the bottom of the panel. They can be different if the submitter chose a colleague as a contact because, for example, a newly hired person without a telephone wanted to register their certificate. You can only view the certificate information on the bottom of the panel. If it is wrong (because, for example, the client chose the wrong client certificate from his key ring), you need to deny the request and ask the client to resubmit a corrected request.

– The remaining fields are:

  • User Name: Frederic Dahm

    This field is the same as the one with the same name on the certificate approval panel. The name to locate the person in the Directory can be changed from this field. This enables the administrator to change it to match one of the entries in the User name field of the user's Person document. The submitted entry can be different if the certificate had a slightly different spelling or format, but this should not invalidate adding it to the Domino Directory. Again, you can select the down arrow to the right of the field to open the Domino Directory (any directories known to your client) to search for names and select the desired name.

  • Send a notification e-mail to the requestor: No

    This option enables you to select that an automatic e-mail will be sent to the requestor. It is selected by default if you selected this option in the CA profile. Note that users should be cautioned that, even when their request is approved, there might be a delay before their certificate is added to the Domino Directory and, when added, propagated to all replicas of it.

  • Reason: <>

    If you need to deny the request you can fill out this field. It will be saved with the request for future reference and included in the e-mail to the requestor.

3. If you approve of the request, click **Approve** in the Action section. To deny the request, click **Deny** in the Deny section. Provide a reason in the Reason field before denying the request. You can check that the addition was successful by checking the requestor's Internet Certificates in their Person document in the Domino Directory, by using the **`ldapsearch`** utility to list their entry, or by locating the request in the Administration Requests database as described in "Step 2: Approve a client certificate request in the Domino CA" on page 196.

This completes the client certificate registration.

D

Troubleshooting policies

The subject of policies has been a difficult subject to grasp for many Lotus Notes administrators and even more difficult to implement correctly. While policies can be used can be used as very effective tool, Notes administrators must have a thorough understanding of how to create and apply them to their organization.

This is why we took the time to explain to explain policies and policy settings in the previous IBM Redbook in this series, *Lotus Security Handbook*, SG24-7017, and explain them in the present publication. We build on the initial knowledge and shown new information that should further help Notes administrators gain a firm grasp on the concept of policies and policy settings.

However, it would have been remiss on our part not to have an extra appendix that discussed ways of troubleshooting problems that can occur with policies, specially if policies are applied incorrectly within the organization.

In this appendix, we discuss how the policy functionality works client-side and where the policy information is written and what can be done to correct incorrect configurations. We start by discussing the tool responsible for retrieving from the server the policy settings that are part of established policies and applying them client-side.

The Dynamic Client Configuration tool

The Dynamic Client Configuration (DCC) tool is a Notes client process that synchronizes certain information between Notes clients and Domino servers. The DCC executable, ndyncfg.exe, is in the Notes client program directory.

The DCC does a lot of work. To begin with, DCC populates the Client Information section on the Administration tab of Person documents. The DCC is also required for the proper operation of certain domains processes such as “Move Mailfile” and new Notes/Domino 6.x features including policies and roaming users. Therefore, if you encounter issues with any of these processes/features, first troubleshoot the DCC.

The DCC runs when the user authenticates with their home server, and either their Person document has been modified, or their assigned desktop policy has been modified since the last authentication. Specifically, during the user's first authentication to the server, the server dynamic profile is compared with the client dyninfo object, which is stored in the Personal Address Book preferences. If there are differences between the dynamic profile and the dyninfo object, DCC runs. Otherwise, the DCC will not run. Technically, ndyncfg.exe can be forced to run by typing **`ndyncfg`** at a MS-DOS® command prompt, but this is not the recommended method of running DCC manually. We discuss this more later in the appendix.

The DCC is designed as a push mechanism only from the server to the client. The DCC updates settings on the user's workstation based on the current settings in the user's Person document and any desktop policies that are in place. For example, if changes are made to a user's Person document, the DCC will detect the changes when the user connects to the server and then push the appropriate changes down to the client. By default, the DCC is installed with every client and runs daily at the first user authentication with the server.

When the DCC executes, it adds the following lines to an entry in the Miscellaneous Events view of the local LOG.NSF:

```
11/06/2005 07:40:00 AM  Dynamic Client Configuration started
11/06/2005 07:40:02 AM  Initializing Dynamic Client Configuration
11/06/2005 07:40:03 AM  Dynamic Client Configuration updating policy information
11/06/2005 07:40:03 AM  Dynamic Client Configuration updating location information
11/06/2005 07:40:03 AM  Dynamic Client Configuration shutdown
```

If this information cannot be found, this is an indication that there is a problem with the DCC. If no errors are reported in the log, this indicates that the DCC ran on the client. However, do note that the log entry only ensures that DCC ran; it does not ensure that the DCC successfully changed any values on the client. For example, the previous log entry that reads `Dynamic Client Configuration updating location information` does not necessarily mean that the location information was successfully updated.

There are also other ways to determine if the DCC is not working. Another place to look in the Domino Directory (that is, NAMES.NSF). There should be client information on the Administration tab of each Person document. If that information is missing, or the information is there but not up to date, there might be a problem with the DCC. In addition, if the policies in place, specifically the desktop policies, seem to skip certain people, that might indicate a problem with the DCC. This also applies to roaming users and mail file moves through adminp.

If problems are identified with the DCC, you need to determine what prevents the DCC from working properly. One common cause for the DCC not working as expected is that it might have been deliberately or inadvertently disabled. Even though the DCC was originally introduced in Release 5 of Notes/Domino, it was not required for many features. Therefore, users and administrators might have disabled it.

If you determine as part of a comprehensive troubleshooting process that the DCC is not running, perform the following steps to make it work properly:

1. Check the NOTES.INI configuration file on the user's workstation, and if the parameter `DisableDynConfigClient=1` is present, remove it.
2. Access the properties of the user's current Location document (click the **Location** section on the status bar, select the **Edit Current** pull-up menu on the Status bar, and then select **File** → **Document Properties**). On the Fields tab, look for the "AcceptUpdates" field. If this value of this field is set to "0" (zero), enable the DCC by performing the following steps:
   a. Open the current Location document.
   b. Select **Actions** → **Advanced** → **Set Update Flag**.
   c. When the prompt "Allow administrators to keep this location's settings up to date with those settings on your mail server" opens, click **Yes**.
   d. Save and close the location document.

   If after performing the previous steps, the DCC still does not work, remove the address book preferences in the user's Personal Address Book (that is, the client copy of NAMES.NSF) using the following steps:
   a. Open the user's personal Name and Address book.
   b. Select **Actions** → **Remove Address Book Preferences**.

You might wonder what the address book preferences have to do with the DCC. When you select the Remove Address Book Preferences option, this removes the directory profile document (directoryprofile), which contains something called $DynInfoCache. With this document deleted, the cache is completely rebuilt when users re-authenticate with their home server. In addition, this basically de-synchronizes the client dyninfo object with the server dynamic profile, forcing the DCC to run on the client's next authentication with its home server.

Note that after removing address book preferences, the users need to reset certain items if they have customized the preferences of their personal Name and Address Book, such a defining the group sort order, the format of contacts, and the address format.

Policy profiles and documents in the $Policies view

The first item written is a policy profile, the second is a collection of one or more documents (based on the number of policy settings the applied policy contains) in the $Policies view.

For policy profiles, these are documents that serve as data and time stamps to specify at what time a specific policy was applied. This serves as a means for the DCC tool to know whether there has been an update in the specific policy and whether the contents of the policy (and related policy settings) need to be re-applied to the Notes client.

Because, by definition, profile documents do not appear in any view (and thus, make it difficult to access them other than through programmatical means), the best tool to aid in accessing policy profiles is NotesPeek. For a free copy of the tool, visit the Lotus Sandbox at the following Web page:

`http://www.lotus.com/ldd/sandbox.nsf/0/2791869f4e1d3fa385256f2c00432973?OpenDocument`

Using NotesPeek, the user's local Name and Address Book (that is, the local NAMES.NSF database) can be opened and all profiles document can be accessed. These are conveniently

grouped under the Profiles category, as shown in Figure D-1, which shows not only policy profiles (identified as $policyprofile) but other profile documents as well.

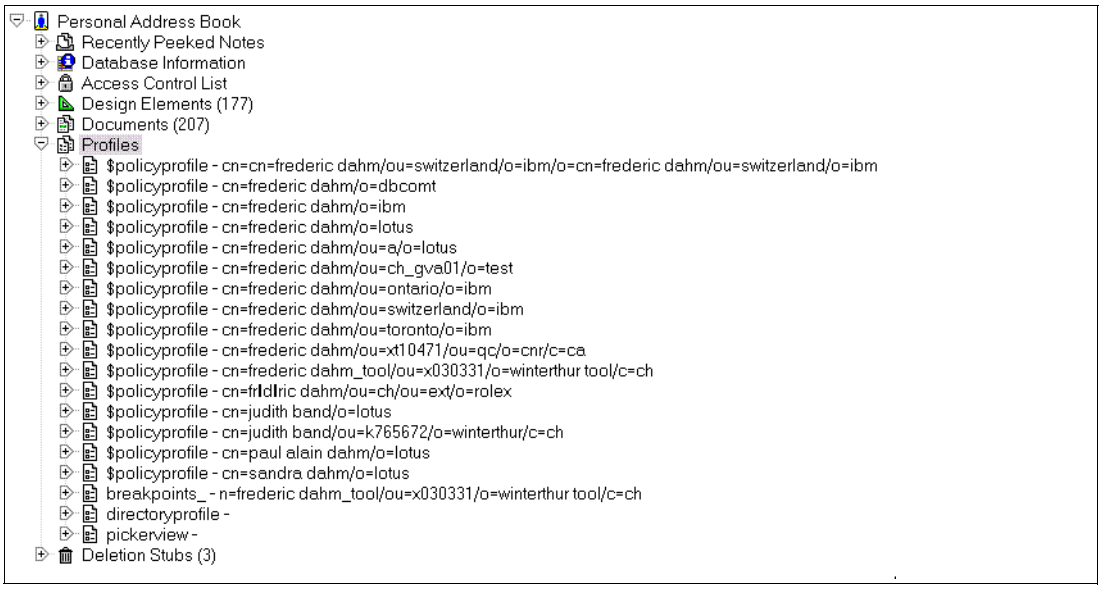


Figure D-1 Example of the collection of accumulated policy profiles

To clear these policy profiles, use the code in Example D-1 in a button, action button, or agent.

Example: D-1 Clear policy profiles code

```
Sub Initialize

   '--- Declare Class Variables
   Dim s As New NotesSession
   Dim db As NotesDatabase
   Dim doc As NotesDocument

   '--- Initialize Class Variables
   Set db = s.currentdatabase
   Set col = db.GetProfileDocCollection("$policyprofile")

   '--- Remove all the Policy Profiles
   Call col.Removeall(True)

End Sub
```

The code, in effect, removes all policy profiles (but not the other profiles, which is why it is best to use this code) in the user's local Name and Address Book. Figure D-2 on page 211 shows the result of using this code.

Figure D-2 Example of the policy profiles collection after cleanup

Thereafter, when the user connects again to a server that contains a policy (either an organizational policy for the hierarchy in which the user finds himself in or an explicit policy which has been assigned to the end user), the DCC brings down the policy settings to the Notes client and records the operation in a policy profile, as shown in Figure D-3.

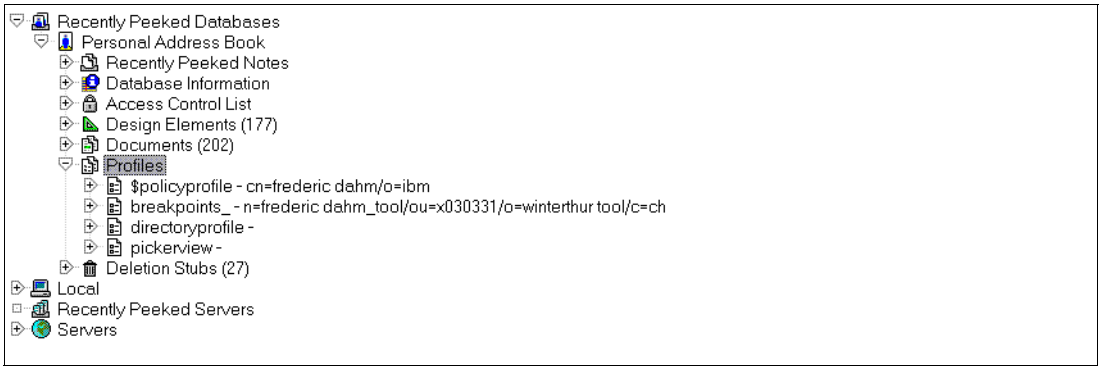


Figure D-3 Example of the collection after the application of a policy

Policy documents

The other place that the DCC records information is in the $Policies view of the user's local Name and Address Book. This is a hidden view, so you need to use a special trick to access the view. Go to the Notes workspace (the page with all the database placeholders). Press the Ctrl+Shift (and keep them pressed), right-click the local Name and Address book database placeholder, and select **Database** → **Go To**. This opens a list of views in the database, including the hidden views. Select the **$Policies** view. This opens the view, similar to the one shown in Figure D-4 on page 212.

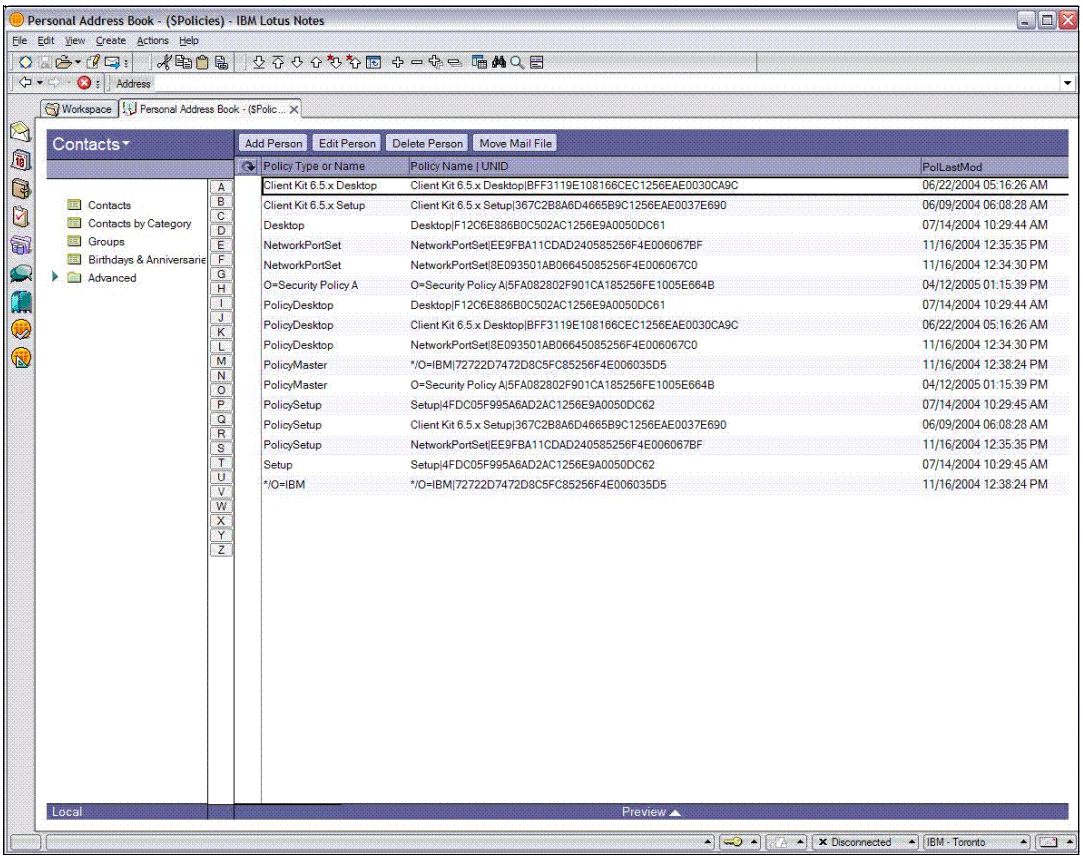


Figure D-4 The $Policies view with a collection of documents

This is another place where there might be problems and, as for the policy profiles, it is best to use the code in Example D-2 for a button, action button or agent.

This code removes all documents in the $Policies view in the user's local Name and Address Book. After the user reconnects, the effective policy defined for this user (where it is organizational or explicit) will be brought down and entries written in the user's local Name and Address Book. Figure D-5 on page 213 shows the result of using this code and the writing of the new policy information.

Example: D-2 Remove all documents in $Policies view code

```
Sub Initialize

   '--- Declare Class Variables

   Dim s As New NotesSession
   Dim db As NotesDatabase
   Dim view As NotesView
   Dim doc As NotesDocument
   Dim collection As NotesViewEntryCollection
   Dim entry As NotesViewEntry

   '--- Initialize Class Variables

   Set db = s.GetDatabase("", "names")
   Set view = db.GetView( "($Policies)" )
   Set collection = view.AllEntries
   Set entry = collection.GetFirstEntry()
```

```
   '--- Process all documents in the ($Policies) view

   While Not(entry Is Nothing)
      Set doc = entry.Document
      doc.Remove(True)
      Set entry = collection.GetNextEntry(entry)
   Wend

End Sub
```

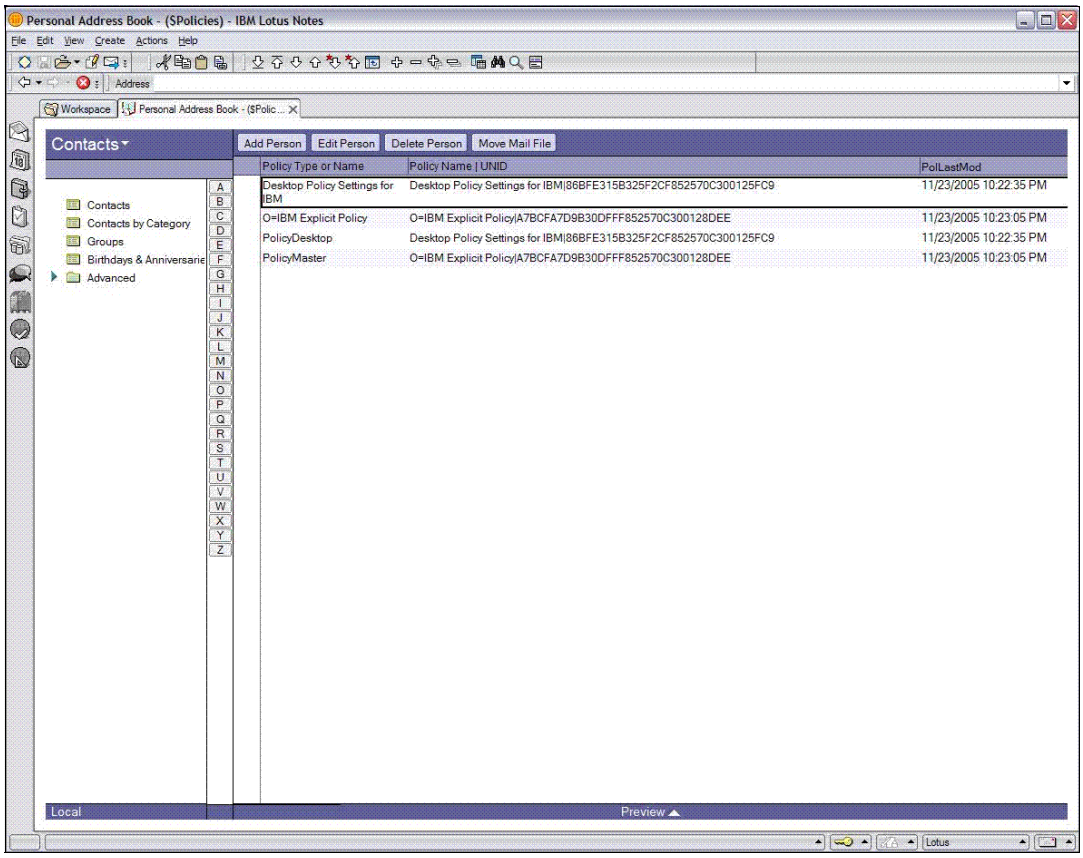


Figure D-5 The $Policies view cleaned up and with the application of a policy

In Figure D-5, there was an explicit policy defined for the user, which contained a desktop settings document. When bringing down the settings for the policy, a total of four documents were written, one for the policy itself and three documents related to the desktop policy settings.

The cleanup procedure

We explained the Dynamic Client Configuration (DCC) tool, the policy profiles, and the documents in the $Policies document. We also explained the method by which to clean the policy profiles and the documents in the $Policies view.Now, we describe the actual cleanup procedure if a problem exists with one user.

Assuming that the DCC works as expected (there is rarely, if ever, a problem with the DCC), the way to correct policy application problems for the user is to clean up both the policy profiles and the documents in the $Policies view, which are all in the user's local Name and

Address Book. It is best to clean up both of these, because the problem can persist if only one is cleaned up.

To save time and effort (and to reduce the possibility of an error), use the code written for the purpose of cleaning up both the policy profiles and the $Policies view, as shown in Example D-3.

Example: D-3 Clean up policy profiles and $Policies view code

```
Sub Initialize

   '--- Declare Class Variables

   Dim s As New NotesSession
   Dim db As NotesDatabase
   Dim view As NotesView
   Dim doc As NotesDocument
   Dim collection As NotesViewEntryCollection
   Dim entry As NotesViewEntry

   '--- Initialize Class Variables

   Set db = s.GetDatabase("", "names")
   Set view = db.GetView( "($Policies)" )
   Set collection = view.AllEntries
   Set entry = collection.GetFirstEntry()
   Set col = db.GetProfileDocCollection("$policyprofile")

   '--- Process all documents in the ($Policies) view

   While Not(entry Is Nothing)
      Set doc = entry.Document
      doc.Remove(True)
      Set entry = collection.GetNextEntry(entry)
   Wend

   '--- Remove all the Policy Profiles
   Call col.Removeall(True)

End Sub
```

This code removes all policy profiles and all documents in the $Policies view in the user's local Name and Address Book. When the user reconnects, the effective policy defined for this user (where it is organizational or explicit) is brought down and entries written in the user's local Name and Address Book.

At that time, policies should work properly for this user and the problem should be resolved.

E

Encrypt delivered incoming mail

Since Lotus Domino 5, Domino organizations have had the ability to enforce secure storage of delivered incoming mail either on a per-user[1] or per-server-configuration[2] basis. Considering that interest in broad-based organizational e-mail security continues to grow, and because new Notes C APIs have been released that can allow IBM Business Partners and ISVs broad-based access to ID file management (see Appendix A, “Notes C API security enhancements” on page 155), a short elaboration of this feature’s functionality is useful.

Before an unencrypted e-mail message is delivered to a mail (or mail-in) database, the Domino router checks whether the message should be encrypted to the user’s public key. If it is to be encrypted, the type of encryption applied depends on the format of the message’s body content. If the format is Notes rich text, conventional Notes encryption is performed. By contrast, if the format is MIME and an X.509 certificate for the recipient is available, trusted (cross-certified), and unexpired, S/MIME encryption is performed. However, if in the MIME format an appropriate X.509 certificate is unavailable (not present, untrusted, or expired), *conventional Notes* encryption is performed while still maintaining the MIME format of the message (that is, the message is not converted to Notes rich text).

[1] Go to the Domino Directory → People view → Basics tab → Mail section → When receiving unencrypted mail, encrypt before storing in your mailfile field.

[2] Go to the Domino Directory → Configuration\Servers\Configurations view → Router/SMTP tab → Restrictions and Controls tab → Delivery Controls tab → Delivery Controls section → Encrypt all delivered mail field.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information about ordering these publications, see “How to get IBM Redbooks” on page 219. Note that some of the documents referenced here may be available in softcopy only.

- *Lotus Security Handbook*, SG24-7017
- *A Secure Portal Using WebSphere Portal V5 and Tivoli Access Manager V4.1*, SG24-6077
- *Lotus Notes and Domino R5.0 Security Infrastructure Revealed*, SG24-5341
- *The Domino Defense: Security in Lotus Notes 4.5 and the Internet*, SG24-4848
- *A Secure Portal Extended With Single Sign-On*, REDP-3743
- *Domino Web Access 6.5 on Linux*, SG24-7060
- *iNotes Web Access Deployment and Administration*, SG24-6518
- *Lotus Domino 6 spam Survival Guide for IBM @server*, SG24-6930

Other publications

This publication is also relevant as a further information source:

- Benz, B. et al., *Lotus Notes and Domino 6 Programming Bible*, Wiley, 1st edition, 2003, ISBN 0764526111

Online resources

These Web sites and URLs are also relevant as further information sources:

- IBM developerWorks, Lotus developer resources

  http://www.lotus.com/ldd
- Lotus Education On Demand: Lotus Domino Certification Authority Tutorial

  http://www.ibm.com/support/docview.wss?rs=463&uid=swg27006424
- Lotus Sandbox: NotesPeek program

  http://www.lotus.com/ldd/sandbox.nsf/0/2791869f4e1d3fa385256f2c00432973?OpenDocument
- Lotus Domino 7 Administrator Help

  http://www.lotus.com/ldd/doc/domino_notes/7.0/help7_admin.nsf/
- *Single Sign-on in a Multi-directory World: “Never say login again”*

  http://www.ibm.com/developerworks/lotus/library/sso1/

- The CA Process In Notes/Domino 6

  http://www.ibm.com/support/docview.wss?rs=463&uid=swg27006424
- “Email Messages Addressed to “postmaster” or “abuse” are Being Delivered to the Domino Server Administrator,” Technote 1106677

  http://www.ibm.com/support/docview.wss?uid=swg21106677
- “How to Stop Incoming Mail Addressed to Just the Last Name,” Technote 1090405

  http://www.ibm.com/support/docview.wss?uid=swg21090405
- “SMTP Mail Is Received by User in Spite of User's Different Internet Address in Person Document,” Technote 1192804

  http://www.ibm.com/support/docview.wss?uid=swg21192804
- dotNSF

  http://dotNSF.com
- SearchDomino Web site

  http://searchdomino.techtarget.com/
- Domino Security.org, a Web site that helps you understand and implement the security features in Lotus Domino and Notes

  http://www.dominosecurity.org/
- RSA Laboratories, PKCS #11: Cryptographic Token Interface Standard

  http://www.rsasecurity.com/rsalabs/node.asp?id=2133
- The Spamhaus Project

  http://www.spamhaus.org
- The Anti-Phishing Working Group (APWG)

  http://www.antiphishing.org
- The Web Robots Pages

  http://www.robotstxt.org
- Email etiquette

  http://www.emailreplies.com
- SPF: Sender Policy Framework

  http://www.openspf.org
- Yahoo! Anti-Spam Resource Center, DomainKeys

  http://antispam.yahoo.com
- OpenNTF: Surely Template

  http://www.openntf.org
- Internet Systems Consortium, Inc. (ISC), BIND software

  http://www.isc.org/
- rbldnsd: Small Daemon for DNSBLs

  http://www.corpit.ru/mjt/rbldnsd.html
- djbdns

  http://cr.yp.to/djbdns.html
- PGP Corporation

  http://www.pgp.com

- Penumbra Group

  http://www.PenumbraGroup.org
- RHS Consulting

  http://www.rhs.com
- STDI Consulting Inc.

  http://www.stdi.com
- CHC-3

  http://www.chc-3.com
- DotNSF

  http://www.dotnsf.com
- CODEX Software A.G.

  http://www.icodex.com

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

Numerics

A

B

C

D

E

F

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

Z

(0.2”spine)
0.17”<->0.473”
90<->249 pages

Security Considerations in Lotus Notes and Domino 7:

Making Great Security Easier to Implement

Redbooks

New security features in Lotus Notes and Domino 7

Best practices for implementing Lotus Notes and Domino security

Applicable scenarios and examples

Strong security has always been part of the family of Lotus software products. More notably, it has been a feature that has made Lotus Notes and Domino an industry leader for security-rich messaging, calendar, and scheduling capabilities. with a robust platform for collaborative applications. With Lotus Notes and Domino 7, IBM extends the reach of Lotus Domino messaging and collaboration solutions while continuing to leverage your IT and application investments. The new version offers capabilities to support more people with fewer servers, to simplify administration, and to provide tighter integration with Web standards.

In this IBM Redbook, we discuss specific security and anti-spam enhancements that have been incorporated into Notes and Domino 7.0.

This publication is the fourth in a series about IBM Lotus security to be published. The previous IBM Redbooks about the topic are, in chronological order, *The Domino Defense: Security in Lotus Notes 4.5 and the Internet*, SG24-4848, *Lotus Notes and Domino R5.0 Security Infrastructure Revealed*, SG24-5341, and *Lotus Security Handbook*, SG24-7017.

The primary goal of these publications was to focus on the strong security that has always been part of the family of Lotus products. The present publication continues down the path set by these previous Redbooks, offering, as with each previous release, information about key features and functionalities pertaining to the security aspects of Lotus Notes and Domino Release 7.0.x, as well as best practices to implement these new features and functionalities.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-7256-00 ISBN 0738497347