Global Information Assurance Certification Paper

Lotus Notes Penetration

GCIH Practical Assignment

Version 2.0 (revised August 13, 2001)
Option 1

By Karl Rademacher
Final Draft: December 26, 2001

SANS NEWSBITES

The SANS Weekly Security News Overview

Volume 2, Number 27 July 5, 2000

-- 29 June 2000 - Intel Employee Pleads Guilty

A former Intel employee has pleaded guilty to computer fraud; when he was fired from his job three years ago, he logged on to the company's internal system from his home computer and deleted files which slowed the company's chip manufacturing process for several hours.

[Editor's note: (Grefer) This is another reminder that most costly attacks still come from insiders; with disgruntled staff being one of the prime originators.]

Part 1: The Exploit

We've seen it before: A hacker, mustering all the gloat he can, makes a breathless announcement that will shatter our perception of secure e-mail forever. Alas, a nasty security hole has been punched through the armor known as Lotus Notes. Script kiddies and other evildoers will soon be provided the tools to bum-rush your company's prized information gold mine with utter abandon. Corporate officers will bear witness to Mountain Dew swilling barbarian hoards plundering all they hold near and dear. Top Secret CIA documents will be auctioned off on Ebay to people with names like "SaddamsBuyer247." Cats and dogs will… wait!

Exploit Basics

Vulnerability: Loose Access Control
Target: Lotus Notes / Domino.
Target Type: Enterprise E-mail, information management and groupware application.
Versions Affected: 4.1 through 5.09
Operating System: N/A
Description: By taking advantage of default or poorly managed ACLs, it is possible to compromise entire Domino mail Infrastructures.
Protocol: TCP, using ports 80 (web) and 1352 (Notes)
Tools Used: Web Browser and Notes Client
Variants: None
Related Information:
At Security Focus
At @Stake (L0pht)

Truth be told, the lion's share of Domino vulnerabilities are really nothing new. Most of them mention gaping holes in the Notes Access Control List (ACL) or Notes Execution Control List (ECL) as if they were horrible oversights by Lotus. The fact is, the default security levels in Domino aren't true vulnerabilities, but dubious triumphs of convenience over security.

A summer 2000 article in 2600: The Hacker's Quarterly, planted the seed that eventually grew to become "The Incident" described herein. In the article, the writer gave a brief treatise on compromising Notes systems. After reading it, I decided that it would be good to test the armor at my place of employment before some evildoer did.

Specifically, I wanted to determine just how vulnerable my employer, Super Duper International (SDI), was to disgruntled ex-workers. After all, it was no secret that SDI's IT department had an incredibly high turnover rate. Experienced workers were bailing out for better paying jobs elsewhere, and SDI was going through "yet another purge" of consultants and "dead wood."

To put myself in the right frame of mind, I imagined that I was fired for prolific web surfing during business hours. I didn't deserve to lose my job and I wanted to get even. Super Duper International needed to be hacked! Better yet, I would steal confidential internal company information, and sell it to a competitor. After that, I'd leave a trail of destruction that would take years to clean up. That will show them!

Tools Used

This incident involved my very first foray into the wilds of Whitehat testing. In addition to securing full permission from management to execute this probe, I refused to use any sort of script kiddie pre-packaged tool. I leveraged basic knowledge gleaned from four solid years of administering Notes servers and absorbing knowledge from other administrators. While I'm a Lotus Certified P-CLP[1], I wasn't (and still am not) an elite hacker, so the software used wasn't anything special: Internet Explorer 4.01, Super Scan 2.1, Windows NT nslookup, and Lotus Notes 4.67. Had I to do it again today, I'd also use nmapNT, newer versions of the software listed above, DominoScan, Sam Spade, and IPR from Patrik Karlsson.

Part 2: The Attack

Deciding Upon a realistic target

The people responsible for SDI's US-based firewalls and IDS sensors were an honorable and diligent lot. Each of them held enough certifications to fill both sides of an average business card in 2-point font. Since SDI-USA fed them a regular diet of Red Bull and moon pies, none would dare leave. Add to this the fact that I knew nothing about punching through a well-managed firewall, and you get a prime recipe for failure. So, rather than waste my time whacking away on the front door, I looked for another entry. After all, SDI, being a typical multi-national corporation, didn't have the entire WAN governed by just one group.

In fact, Super Duper International had numerous subsidiaries throughout the world. Most employees (and many outsiders) knew that these business units were all connected via some common internal WAN. Additionally, it was no secret that several of our overseas operations bore the burden of tremendous budgetary constraints due to tight economic conditions. It was easy to imagine that some things – like security – may have been overlooked in the interest of saving money. So, How did I chose a target? Well, I saw many reports on servers getting defaced or otherwise compromised in Brazil via the Attrition[2] and Safemode sites. Assuming security in general was pretty weak there, that's where I went first. E-Mail systems happen to be my forte, so I concentrated my search on SMTP MX records. Nslookup revealed a potentially vulnerable Brazilian SuperDuper International Subsidiary with direct service to the Internet:

[1] Principle Certified Lotus Professional
[2] The defaced Website portion of Attrition is now defunct.

```
>set type=mx
>superduper.com.br
Server:  dns03.myspecialdnsserver.net
Address:  64.x.x.x

SuperDuper.com.br    MX preference = 1, mail exchanger = mx.SuperDuper.com.br
SuperDuper.com.br    MX preference = 2, mail exchanger = mx.otherdomain.com.br
SuperDuper.com.br    nameserver = dns1.otherdomain.com.br
SuperDuper.com.br    nameserver = dns2.otherdomain.com.br
mx.SuperDuper.com.br              internet address = 200.x.x.x
mx.otherdomain.com.br    internet address = 200.x.x.x
dns1.otherdomain.com.br    internet address = 200.x.x.x
dns2.otherdomain.com.br    internet address = 200.x.x.x
```

Anonymous contact

A cursory port scan of mx.SuperDuper.com.br revealed that the server was listening on TCP ports 25 (SMTP), 80 (http), 1352 (Lotus), and 2301 (Compaq Insight Manager). The web banner let me know the machine was running Domino 4.6.4 on Win32. I guessed that perhaps the local admin set up this SMTP gateway for ease of administration while they're at home, and possibly to allow workers the luxury of checking mail via the Internet. My next move was to start Internet Explorer and type **http://mx.SuperDuper.com.br/log.nsf.** The result was the Notes server's log. A click on the link to database usage gave me a list of databases on the system. Looking at the individual listings for each database revealed many interesting things to me. Most importantly, I learned that the Notes group responsible for managing everything is called **SDISA-Admins**.

The log was great, but the Notes Address Book (NAB) was better. The NAB is the cornerstone of security for every Notes installation in the world. This database defines important things like Access control groups, authorized users, key trusts, and server connections. Access to this database should be tightly defined and closely watched. In this case, it was wide open. I accessed it without a password, using Internet Explorer.

Once in the NAB, I clicked the link to Groups. I located **SDISA-Admins**, opened it up, and print it off for later reference. Next stop was the People link, where I poked through links at random, and quickly realized I struck gold. A good portion of the person documents contained downloadable user ID files! For those who don't know, having a Notes ID file is like having someone's private PGP key. Possession of the correct one – like that of a highly placed administrator – puts you way ahead in the battle toward compromising a Notes system. Among the numerous IDs I collected were two members of the **SDISA-Admins** group, and several individuals with important sounding job titles (CIO, Executive Vice President, Chairman, etc). I could have literally gathered hundreds of them, but I was lazy. I settled for 25.

The next step was to poke through the Servers\Servers view, where I harvested three server ID files. Last, I visited the Servers\Connections view, believing it would provide a great roadmap to the SDI Latin America Notes (and probably WAN) infrastructure. It was here that I found out that the smtp server was named Webnotes/BR/SDI.

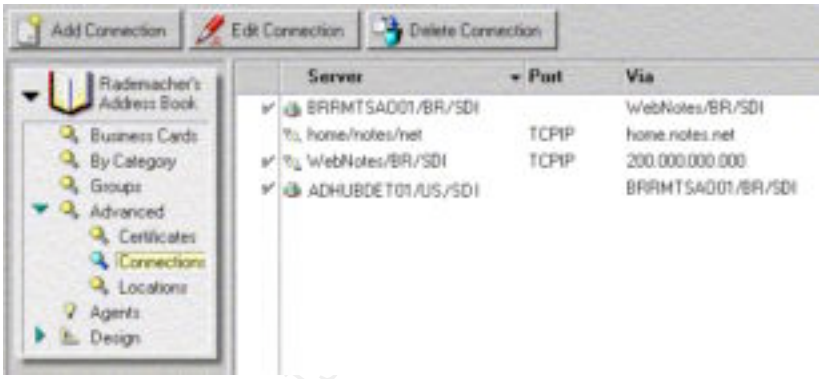
Access Elevation

After that, I started Notes on my local desktop and started manually guessing passwords. On the tenth try, I got the basic pattern (first initial last name, all lower case). I dutifully wrote down the passwords, and commenced to impersonate Raul Fernandez (not his real name), a member of **SDISA-Admins.** At this point, I "owned" SDI's entire South American messaging infrastructure. A thief in my place could have pulled a replica of every confidential marketing and legal database SDI owned, burned them to CD and easily sold them to the nearest gray-market data broker.

The SDI NAB contained a group called "Allowed-to-use-Passthru" (obviously geared toward using Domino Passthru). This caught my attention because I knew how badly passthru could be abused if it wasn't properly configured and maintained[3]. In this case, it was being used to allow employees to connect via the Internet, and get to their mail accounts and other company data, which was housed on servers hidden by the firewall.

Potential for Irreparable Damage

"Raul" edited the "Allowed-to-use-Passthru" group, allowing himself expanded rights to roam freely. However, to exercise these rights myself, I also needed to create some passthru connection documents in my local client address book (shown below):

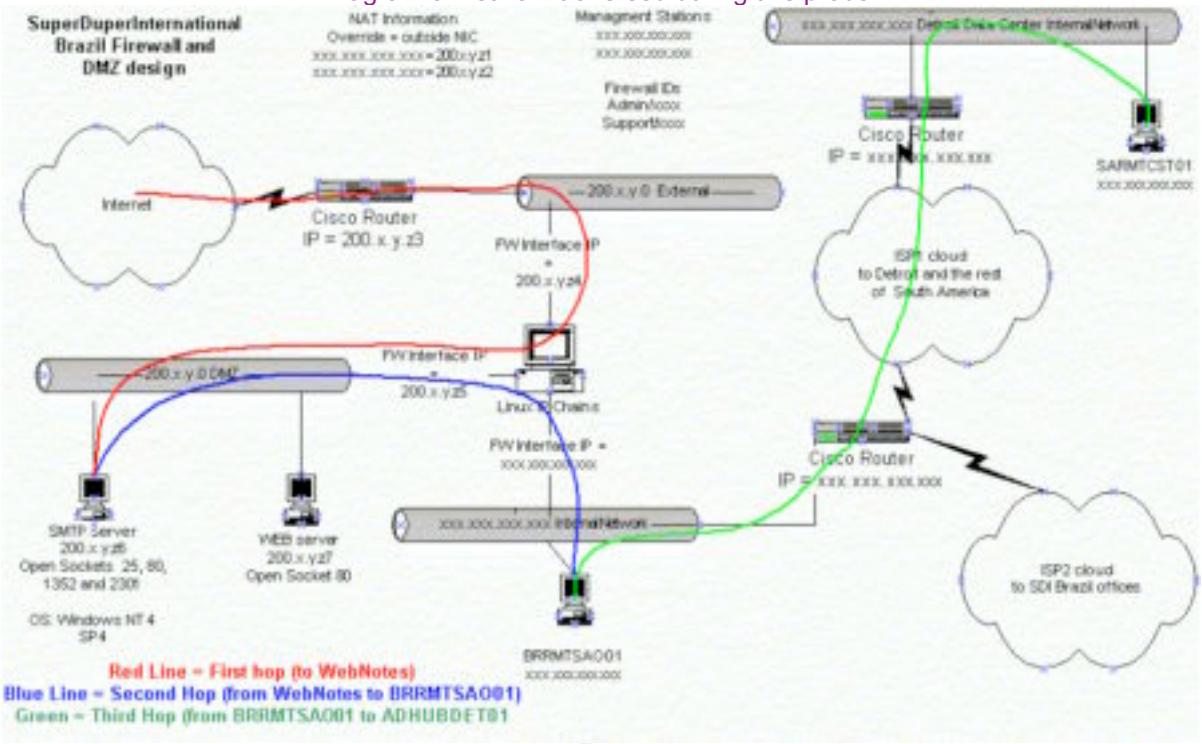


My initial point of entry was WebNotes (line 3). From outside the firewall, I couldn't directly connect to the other Domino servers. However, judging from the Servers\Connections view in the NAB, WebNotes could reach BRRMTSAO01. So, I created the 1st connection document to take advantage of this. BRRMTSAO01 had the connectivity on the internal WAN to get at my real target: ADHUBDET01 – the administrative server for the domain, conveniently located in Detroit, Michigan. The area of the network it resided on was protected by the phalanx of nearly impenetrable US-based firewalls. Other "Bastion" application servers populated

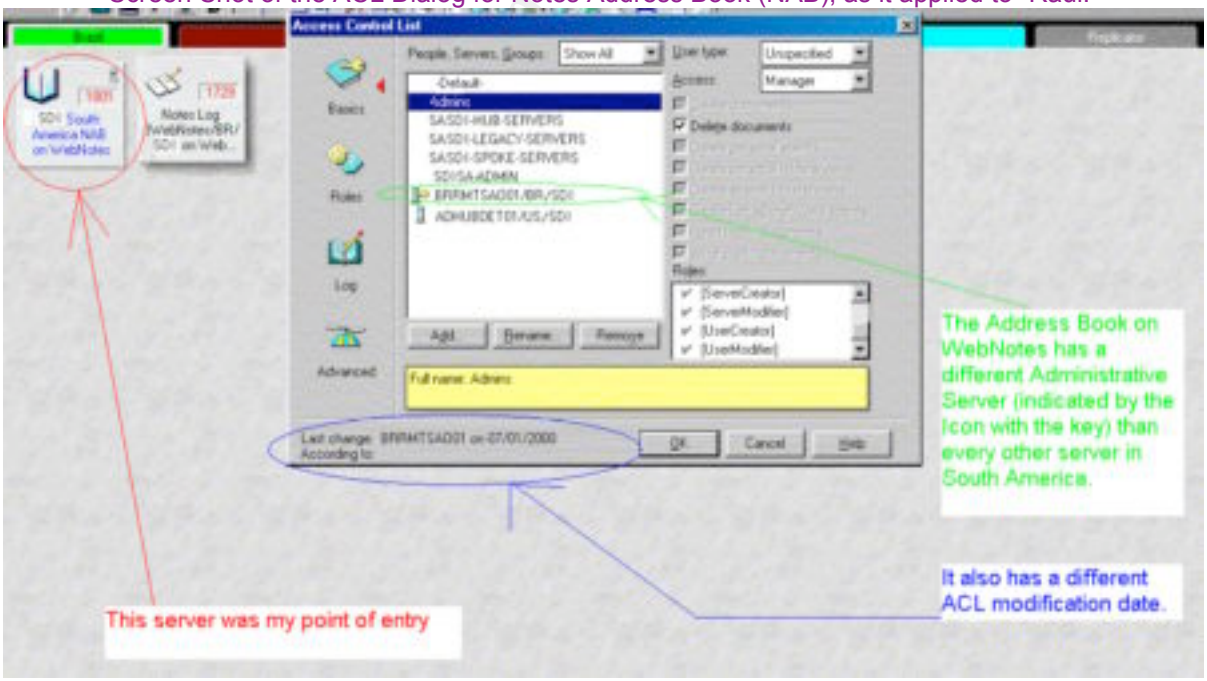
[3] Passthru is a feature which allows users to authenticate on one machine, then hop to others, without having to establish a separate network connection – the first machine does it for you. As long as the destination server is reachable from the original machine AND the user in question has access rights, it works well. It's often used to centralize things like dial-in connections (users only have to dial in to one box from home, and the company has only one modem pool to worry about). By default, passthru has zero logging.

this segment, like PeopleSoft™, some custom Oracle™ applications, and many other sensitive items. This made ADHUBDET01 a very attractive target.

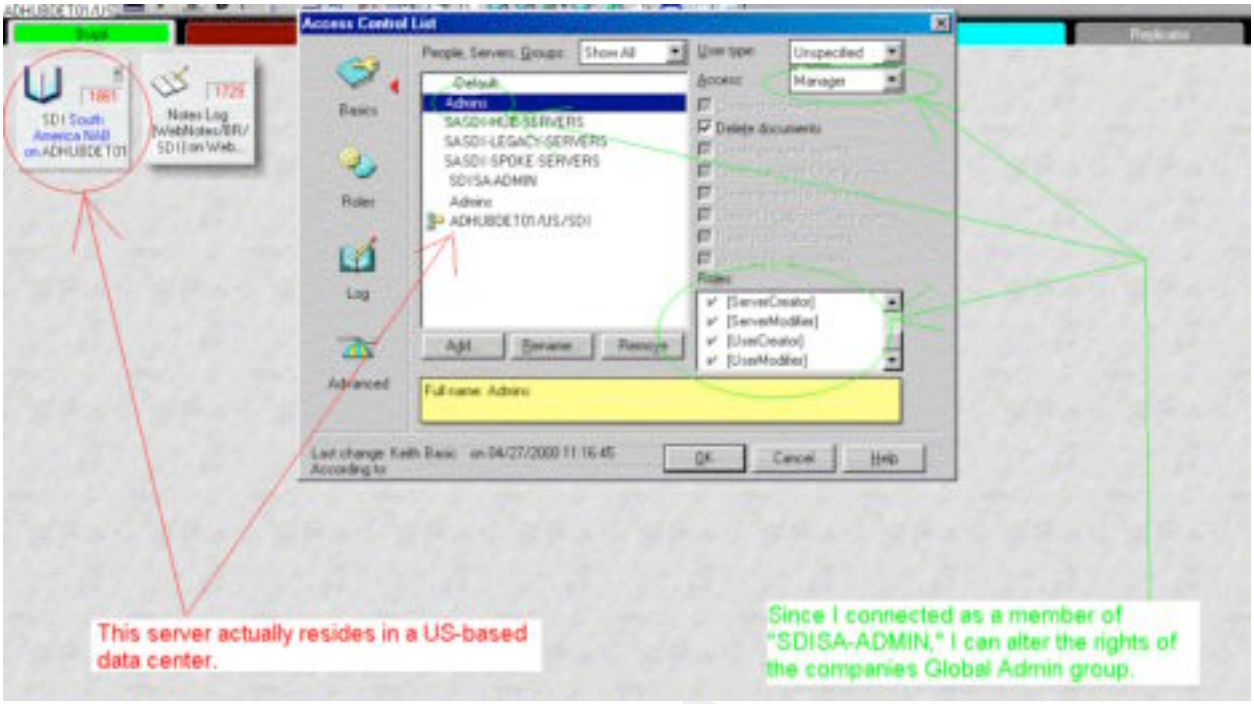
Diagram of network traversed during this probe.



Screen Shot of the ACL Dialog for Notes Address Book (NAB), as it applied to “Raul.”



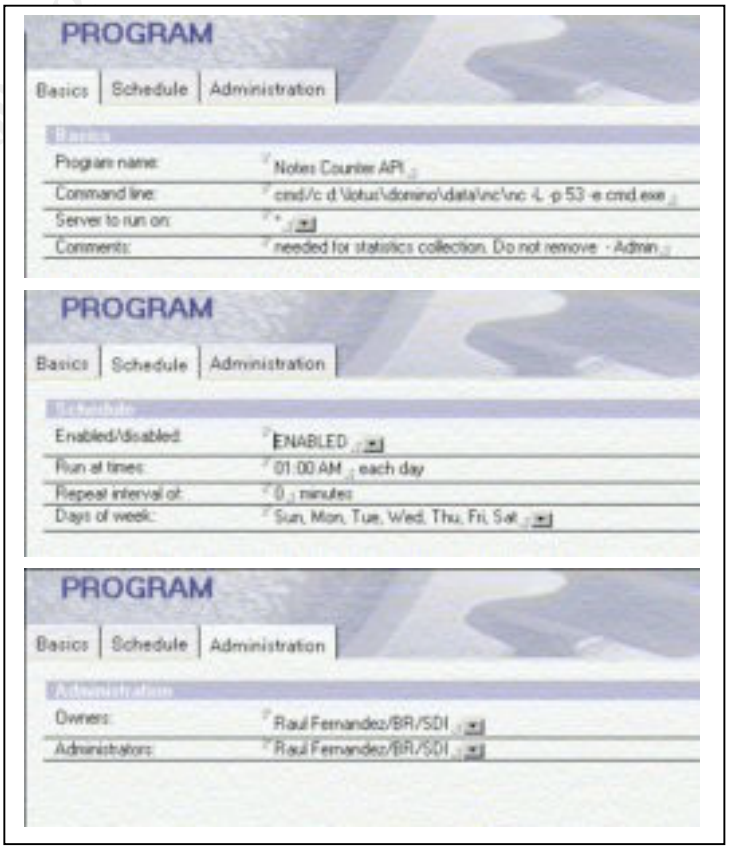
Note how the buttons in the ACL pop-up windows weren’t grayed-out. This meant “Raul” was a Manager, able to add/remove/edit anyone’s rights to the NAB (as well as add/edit/delete all documents within the NAB). Since I was “Raul,” and the NAB governs similar rights to all the other databases, every bit of information within the SDI Notes system was essentially mine to play with.



From ADHUBDET01, someone inclined toward malicious mischeif could have made changes to or stolen information from every Notes server in South America. With relatively little work, they could have done other nasty things as well.

Imagine this scenario:

- "Raul" created a database that contained a copy of Netcat and planted a replica on every server.
- "Raul" then programmed a time-triggered Notes agent set to download nc.exe directly to each server's hard drive every night at 12:45 AM.
- "Raul" finished his handy work by creating a scheduled program entry in the Notes NAB like the example to the right.



At this point, I'd seen and done enough. I collected my notes, boiled them down to something readable, and approached the Domino administrators to let them know what I'd found. They weren't pleased with the news. More on this later...

Snort Signatures

A few weeks after the foray discussed above, I had the opportunity to analyze it in depth. Before replicating this incident, I started a win32 Port of Snort 1.6 without any rules, and minimized it. From the resulting dumps, I developed a few simple signatures, which I submitted to snort.org. I believe (though I'm not 100% positive ☺) they contributed to the following pre-packaged rules:

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC Lotus DelDoc attempt";flags: A+; content:"?DeleteDocument"; nocase;)
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC Lotus EditDoc attempt";flags: A+; content:"?EditDocument"; nocase;)
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC Lotus Domino directory traversal"; content:".nsf/"; nocase; flags:A+;)
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC Domino catalog.nsf access";flags: A+; content:"/catalog.nsf"; nocase;)
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC Domino domcfg.nsf access";flags: A+; content:"/domcfg.nsf"; nocase;)
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC Domino domlog.nsf access";flags: A+; content:"/domlog.nsf"; nocase;)
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC Domino log.nsf access";flags: A+; content:"/log.nsf"; nocase;)
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC Domino names.nsf access";flags: A+; content:"/names.nsf"; nocase;)
```

While writing this practical, I made a few more Snort Rules. These signify deliberate compromise attempts:

```
alert tcp any any -> any 80 (msg:"WEB Domino R4 WebAdmin Template access via replica ID";flags: A+; content:"/852564BD001759F0"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino R5 WebAdmin Template access via replica ID";flags: A+; content:"/852566C90012664F"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino WebAdmin Template Named file access";flags: A+; content:"/webadmin.ntf"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino Notes.ini access attempt";flags: A+; content:"/notes.ini"; nocase;)
alert tcp any any -> any 25 (msg:"SMTP Domino DOS Attempt";flags: A+; content:"bounce@[127.0.0.1]"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino Single dB Denial of Service Attempt";flags: A+; content:"/./"; nocase;)
```

This one flags a systematic vulnerability assessment by DominoScan.

```
alert tcp any any -> any 80 (msg:"WEB DominoScan Assessment of Domino Server";flags: A+; content:"/cgi-bin/nextgenss.exe"; nocase;)
```

These should be considered suspicious, but potentially legitimate.

```
alert tcp any any -> any 80 (msg:"WEB Domino Server-wide Database Listing";flags: A+; content:"/?OpenServer"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino Server-wide Database Listing";flags: A+; content:"/!OpenServer"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino System Events dB access";flags: A+; content:"/events4.nsf"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino Statistic dB access";flags: A+; content:"/statrep.nsf"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino Server mailbox access";flags: A+; content:"/mail.box"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino SMTP Inbound Mail Queue access";flags: A+; content:"/smtpibwq.nsf"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino SMTP Outbound Mail Queue access";flags: A+; content:"/smtpobwq.nsf"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino SMTP Table dB access";flags: A+; content:"/smtptbls.nsf"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino SMTP Table dB access";flags: A+; content:"/mtatbls.nsf"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino WebAdmin Database access";flags: A+; content:"/webadmin.nsf"; nocase;)
alert tcp any any -> any 80 (msg:"WEB Domino Off Line Access Admin dB access";flags: A+; content:"/doladmin.nsf"; nocase;)
```

Domino Log Excerpt

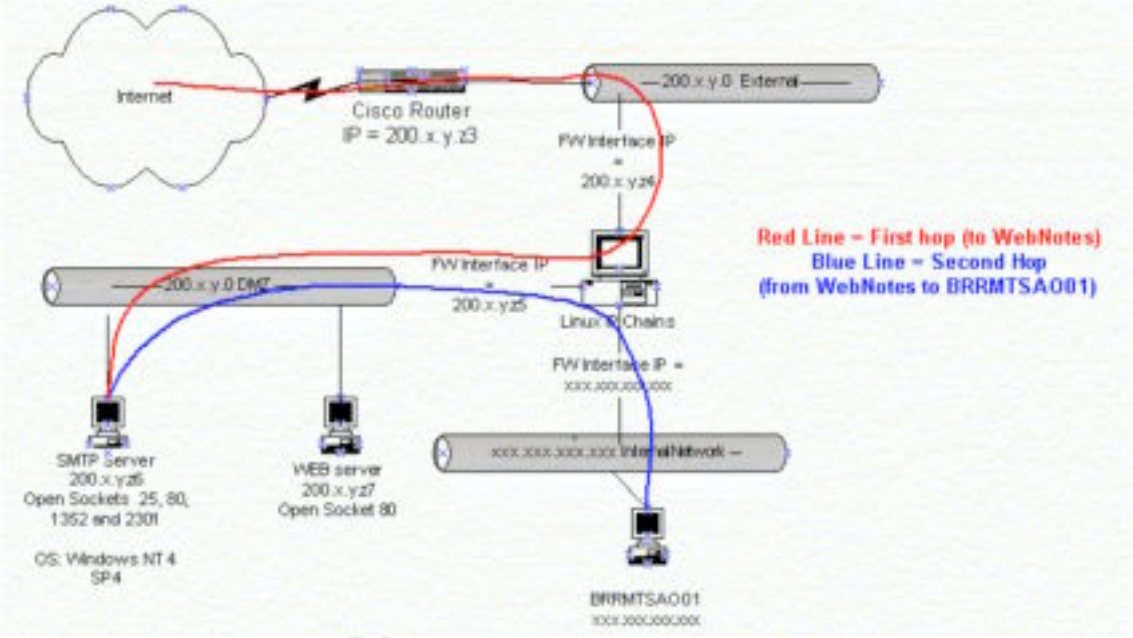
The log excerpt below came from a Domino R5 server that was recently subjected to an information gathering probe much like the one I initially did to mx.SuperDuper.com.br. The Dash ("-") under **User** signifies an anonymous user. **Address** is the client machine's resolved name. See where the **log.nsf** database was accessed at 12:10:40? Then, roughly half a minute later, at 12:11:20, the same client accessed the **names.nsf** file, specifically looking into the **Groups** view, and proceeded to hunt around. Twenty three seconds later (12:11:43), this person opened the **People** view, and opened up various links there as well. I'd seriously worry about this guy.

Date	User	Address	Request
12/13/2001 12:10:40 AM	-	ds1000-001-001.ch1.dsl.	GET /log.nsf HTTP/1.1
12/13/2001 12:10:40 AM	-	ds1000-001-001.ch1.dsl.	GET /log.nsf/$icon?OpenIcon HTTP/1.1
12/13/2001 12:10:44 AM	-	ds1000-001-001.ch1.dsl.	GET /log.nsf/852556420004c6b5885255b040057e525?OpenView HTTP/1.1
12/13/2001 12:10:45 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/previewg.gif HTTP/1.1
12/13/2001 12:10:45 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/nextviewg.gif HTTP/1.1
12/13/2001 12:10:45 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/expview.gif HTTP/1.1
12/13/2001 12:10:46 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/schview.gif HTTP/1.1
12/13/2001 12:10:46 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/colview.gif HTTP/1.1
12/13/2001 12:10:47 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/collapse.gif HTTP/1.1
12/13/2001 12:10:47 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/ecblank.gif HTTP/1.1
12/13/2001 12:10:51 AM	-	ds1000-001-001.ch1.dsl.	GET /log.nsf/852556420004c6b5885255b040057e525/a74e1524df5504693862563260036f557?OpenDocument HTTP/1.1
12/13/2001 12:11:20 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/groups?OpenView HTTP/1.1
12/13/2001 12:11:20 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/actn030.gif HTTP/1.1
12/13/2001 12:11:21 AM	-	ds1000-001-001.ch1.dsl.	GET /domjava/nvapplet.cab HTTP/1.1
12/13/2001 12:11:21 AM	-	ds1000-001-001.ch1.dsl.	GET /domjava/view_en_us.properties HTTP/1.1
12/13/2001 12:11:22 AM	-	ds1000-001-001.ch1.dsl.	GET /domjava/view.properties HTTP/1.1
12/13/2001 12:11:22 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/expand.gif HTTP/1.1
12/13/2001 12:11:23 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/collapse.gif HTTP/1.1
12/13/2001 12:11:23 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/trash.gif HTTP/1.1
12/13/2001 12:11:23 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/vwicn082.gif HTTP/1.1
12/13/2001 12:11:24 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/85255ED5006CAFEF852556D4006CA21C?ReadDesign HTTP/1.1
12/13/2001 12:11:24 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/descsort.gif HTTP/1.1
12/13/2001 12:11:25 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/altdesc.gif HTTP/1.1
12/13/2001 12:11:25 AM	-	ds1000-001-001.ch1.dsl.	POST /names.nsf/85255ED5006CAFEF852556D4006CA21C?ReadViewEntries&PreFormat&Start=1&Navigate=1&Count=40&SkipNavigate=0&SkipCount=0 HTTP/1.1
12/13/2001 12:11:26 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/vwicn004.gif HTTP/1.1
12/13/2001 12:11:26 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/vwicn122.gif HTTP/1.1
12/13/2001 12:11:31 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/85255ED5006CAFEF852556D4006CA21C/D927D1668A5BC00185256926001 4C7D7?OpenDocument HTTP/1.1
12/13/2001 12:11:32 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/people?OpenImageResource HTTP/1.1
12/13/2001 12:11:32 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/actn027.gif HTTP/1.1
12/13/2001 12:11:32 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/actn005.gif HTTP/1.1
12/13/2001 12:11:32 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/actn004.gif HTTP/1.1
12/13/2001 12:11:43 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/People?OpenView HTTP/1.1
12/13/2001 12:11:43 AM	-	ds1000-001-001.ch1.dsl.	GET /domjava/view_en_us.properties HTTP/1.1
12/13/2001 12:11:44 AM	-	ds1000-001-001.ch1.dsl.	GET /domjava/view.properties HTTP/1.1
12/13/2001 12:11:44 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/85255E01001356A8852554C2007531067?ReadDesign HTTP/1.1
12/13/2001 12:11:45 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/ascsort.gif HTTP/1.1
12/13/2001 12:11:45 AM	-	ds1000-001-001.ch1.dsl.	GET /icons/altasc.gif HTTP/1.1
12/13/2001 12:11:46 AM	-	ds1000-001-001.ch1.dsl.	POST /names.nsf/85255E01001356A8852554C2007531067?ReadViewEntries&PreFormat&Start=1&Navigate=1&Count=40&SkipNavigate=0&SkipCount=0 HTTP/1.1
12/13/2001 12:11:53 AM	-	ds1000-001-001.ch1.dsl.	POST /names.nsf/85255E01001356A8852554C2007531067?ReadViewEntries&PreFormat&Start=40.1&Navigate=1&Count=72&SkipNavigate=0&SkipCount=0 HTTP/1.1
12/13/2001 12:11:58 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/85255E01001356A8852554C2007531067/D338985606F8E538852569260014AE2F?OpenDocument HTTP/1.1
12/13/2001 12:11:59 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/people.gif?OpenImageResource HTTP/1.1
12/13/2001 12:12:12 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/ HTTP/1.1
12/13/2001 12:12:13 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/$icon?OpenIcon HTTP/1.1
12/13/2001 12:12:17 AM	-	ds1000-001-001.ch1.dsl.	GET /names.nsf/852560b300643b8a852556c300042c1a?OpenView HTTP/1.1

Prevention at the Firewall

Below is a more focused diagram of the network that I traversed while doing my probe. A firewall rule blocking TCP port 80 to the SMTP server (red line) would have made sifting through its contents with a web browser impossible. I didn't gain access through it, but the version of Compaq Insight Manager (TCP port 2301) on this server was vulnerable to directory traversal attacks. It should have been blocked by the firewall.

The second crossing of the firewall (from the SMTP server to an internal machine – blue line) could have been prevented with a firewall rule that would block the establishment of Notes connections (Port 1352) unless they originate from the internal network. This would require a coordinated effort with the Domino administrator. Mail and Replication connections would have to be initiated at regular intervals by an internal server.



Prevention at the Operating System

Domino requires very little from its host operating system besides an IP stack and a file system. However, it was obvious that WebNotes had plenty of other services running on it. Were they actually needed? Not really. On NT and Win2K boxes, you can permanently disable the following services without effecting **Domino** at all:

- Alerter
- Clipbook Server
- Computer Browser
- DHCP Client
- Directory Replicator
- License Logger
- Messenger
- Netlogon
- Network DDE
- Network DDE DSDM
- Plug and Play
- Remote Procedure Call (RPC) Locator
- SNMP Trap Service
- Server
- Schedule
- Spooler
- TCP/IP NetBIOS Helper
- Telephony

Other Domino server / Win32 Specific rules that I personally use:

1. All file systems **must** be formatted to NTFS.
2. The server should be a stand-alone machine, not a domain controller or a member of a domain.
3. Passwords must be either 7 or 14 characters in length, and contain numbers, punctuation, mixed case letters AND at least one alternate character code (i.e. Press and hold the ALT key, type something like 0252, and release the key).
4. Never leave the password field blank, even for the Guest account (which must be disabled).
5. On Windows NT 4.0 machines, delete the guest account with DelGuest. (It doesn't work on 2K or XP)
6. Do not install Microsoft IIS, SNMP, Simple TCP services or any other extra programs.
7. Delete the terminal service user account and any IIS accounts if they exist.
8. TCPIP should have a static IP address (no DHCP) and **should be the only network protocol installed on your server.**
9. Install the latest security patches for your operating system AND the hardware manufacturer's specific hardware. This includes drivers for custom equipment (like RAID controllers and such).
10. Permanently remove all file shares.
11. Do not install any other operating systems on the computer.
12. If the system BIOS permits, disable booting from floppy or CD. Password protect the BIOS and lock the system case against physical intrusion.
13. Install a server-based firewall. I personally like Black Ice, set to "paranoid," and with advanced firewall rules allowing only those port your server needs to communicate on (TCP 25, TCP 80, TCP 1352, etc)
14. Be diligent! Read your logs on a regular basis.

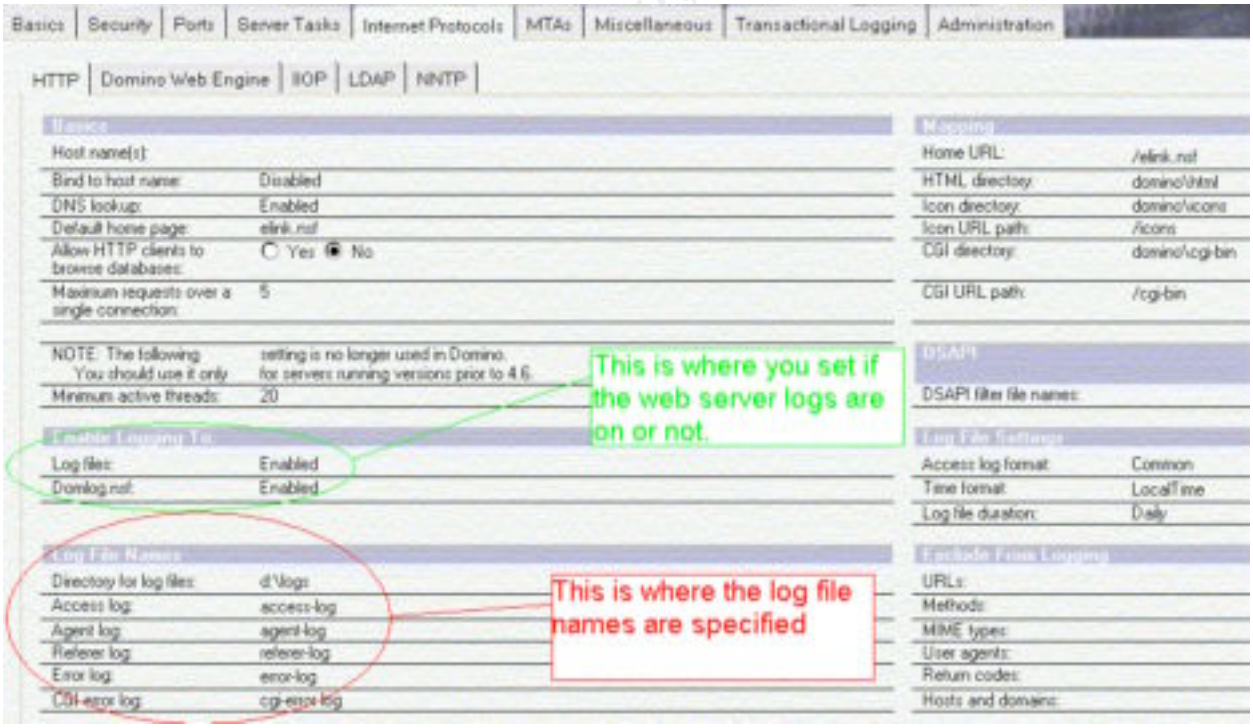
Prevention with Lotus Domino itself

What you can do…

1. If you don't need it, don't run it. If you do need it, make sure you understand it, or find someone who does. The real kicker with the machine compromised in this exercise, is that the web service was not even needed for what the local admin wanted to accomplish (home access to work email via the Notes client).
2. Use the latest version of the software that you can realistically support. As of December, 2001, that is version 5.09.
3. There may be pushback from in-house application developers about major revision changes (going from R3 to R4 or R4 to R5) breaking their applications. Fine, give them a timetable where they'll have time to correct issues, but don't let deadlines

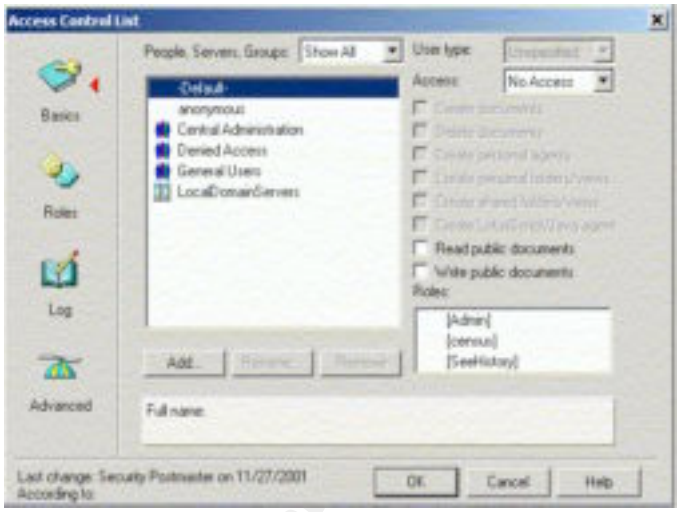
slip. The longer you remain at an older version of Domino, the longer you are exposed to its true programmatic vulnerabilities.

4. Never, **NEVER** allow User or server ID files to be stored in the Address Book. Even with restrictive default ACLs, any person with authorized access to your Notes NAB can download the ID file and guess the password. At the very least, this means your own users can steal each others identity.
5. Remove databases you don't need! To quote from Lotus in security advisory 189425: "…a public Web site should not physically contain any files or data that is not required for its Web applications. This applies to all Web server software. In the case of Domino, we strongly recommend that you remove all unneeded template and database files. For example, once webadmin.nsf has been created, webadmin.ntf can safely be deleted. If a public Domino Web server is replicating with another server behind a firewall, then all the templates can be removed from the public server since the server behind the firewall will apply template changes to its databases and replicate those database changes to the public server."
6. Pay close attention to your logs and statistics reporting databases (domlog.nsf, log.nsf, statrep.nsf, and any text log files specified in the "Internet Protocols\Http\Log File Names" section of your machine's Server document in the NAB (see below). Be especially watchful for odd or abnormal behavior.



7. Subscribe to BugTraq
8. Read Locking Down a Lotus Domino Server, by Andrew G. Hargreave, III
9. Read Hack Proofing Lotus Domino Web Server by David Litchfield.
10. Consider the ACL tips on the following page, in addition to those outlined previously.

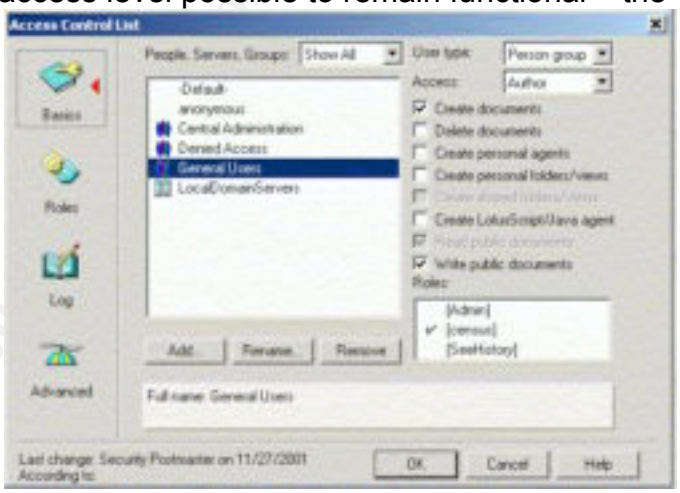
Some Basic ACL Tips



All databases that aren't meant for anonymous public usage should have **Default** and **Anonymous** access both set to **No Access**, like in the picture to the right.

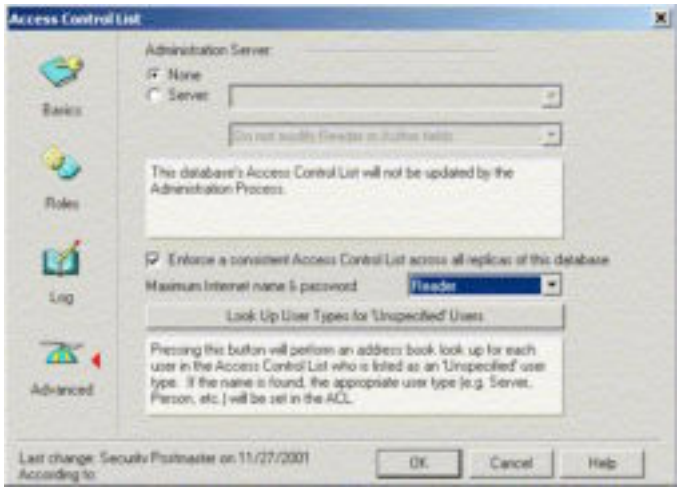
Take the time to create a "Deny Access" group, and list it in every database – including personal mail boxes. It's rights must be **No Access**. After this, make sure all terminated and other known bad users are enumerated in this group.

Non-administrative users (and for non-sandbox environments, **that includes developers**), should get the minimum access level possible to remain functional – the absolute maximum is Editor. Most can get by with Reader or Author.



All groups and individuals listed in the ACL should have their user type defined, where possible. This prevents someone from spoofing access levels by creating a Notes User ID with the same name as a named group.

No matter what the ACL is, if an unauthorized person gets direct file system access to your server, you are sunk. In such situations, The ACL isn't even taken into consideration, unless "Enforce Consistent ACL" is checked.



However, "Enforce a consistent ACL" **is not a panacea**. While it makes ACL changes difficult by unauthorized people or devices, it does not make them absolutely impossible. "Enforce a Consistent ACL" clamps down on casual file browsing by people who have legitimate file system access to your Domino server (like your NT admin). But, it is still possible for a cracker to spoof an ID listed in the ACL and open the database.

One last note on Advanced ACL settings: “The Maximum Internet Access” (R4) or “Maximum Internet Name & Password” (R5) setting should be **reader** (Web Site databases) or **no access** (non-web site databases) for anything that isn’t meant to take anonymous feedback.

What Lotus does

Lotus is actually very good at providing materials to the Domino administrator with an interest in Notes system security. Their security website is fairly comprehensive. They also regularly issue maintenance releases of Domino and make them available at www.notes.net. Visit both sites often!

Other offerings from Lotus you should check out are A Guide to Secure Domino Applications, The Notes.Net September 2001 Special Security Issue of Iris Today, and The IBM RedBook™ on Domino R5 Security

What third parties do

Quite simply, these organizations and companies provide advice, and alerts on potential issues. Some even provide vulnerability scans or scanning software geared toward Lotus Domino. This list is by no means exhaustive.

- Dominosecurity.org provides valuable advice.
- SecurityFocus provides advice, forums and assessment tools.
- The SANS Internet Storm Center and SANS Itself provide security training and advice.
- DomiLockBeta runs a free Domino web vulnerability scanner.
- NGSSoftware, makers of DominoScan
- CERT maintains a list of known vulnerabilities in many software packages.

Part 3: The Incident Handling Process

Preparation

In an Ideal World

Electric excitement filled the air at SDI headquarters in Detroit – the kind of energy felt only during the gravest and most terrifying situations – comparable to the rush a fire fighter feels as lights and sirens clear his way to a four-alarm inferno. SDI had several fully trained and professional security experts at the ready. Our Computer Incident Response Team (CIRT) had been going through drills in anticipation of an intrusion event, and within moments of discovery by our IDS system, a war room was set up, the well-stocked forensic kits deployed and local paid-on-call forensic experts began drilling through the system to find evidence of other intruders. At the same time, our Corporate Information Security Officer (CISO) liaised with the head of Corporate Public Relations

and the two began constructing a press release just in case the absolute worse was discovered. Meanwhile, the lawyers carefully weighed two serious issues: Whether to pursue this fictitious other criminal, and should law enforcement be involved…

What Really Happened?

Had this been a real incident, we would have been caught flat-footed. At the time of my Whitehat probe, the Information Security group at SDI was a young organization striving for recognition. Lean staff conditions precluded time to dig too deeply into any single area. Even less time was available to fully document standardized Incident Handling processes. We were more reactive than proactive.

SDI had a comprehensive set of published security standards, although they were not widely distributed or understood. The general impression was that SDI's Information Security group only dealt with porn surfers, spam filtering, AV software deployment, and pet projects[4]. We were often the last to be contacted when large computing projects were planned by other divisions within SDI.

Did anything prepare SDI for my probe? Amazingly, yes! But it fell short. At the time of this incident, the messaging group was coordinating an International effort to develop standards and define procedures for SDIs Domino infrastructure. This effort covered:

- New server rollout
- Disaster recovery
- Server naming structure
- Certifier naming
- Default database ACLs
- User creation and email database size
- Password strength and ID file storage
- System outages
- Preparing the operating system
- Promoting new applications into production
- Security considerations
- Virus and spam scanning
- System log retention
- Server backups
- Clustering and failover
- And so on.

[4] In the time that has passed since my probe, SDIs security group has undergone major changes. Following a top-down audit of the company by one of the big five, Information Security got reorganized, and was made much more prominent. We now have more of an International "business enabler / service provider" focus. Several new people have been hired, and we've received much needed backing from the powers that be. At least one of the new hires carries a CISSP certification, and three (including myself) are in the process of getting GIAC level 2 certifications. 2001 provided us with able opportunity to cut our teech on large incidents. The SDI Security Group emerged from them all with a greatly enhanced reputation. We also enjoy a much closer relationship to human resources, legal, IT, and upper management than we had previously. Time for training is taken very seriously now. SDI has a true CIRT team, and can, with little notice, do much of the stuff I spoke of tongue-in-cheek on the previous page.

SDIs US Messaging unit had a clearly outlined and easily accessible Notes database that stored all these wonderful standards. Similar things were also done for firewalls, WAN topology, and data centers as well. Unfortunately, it all was done in English. The administrators in Brazil didn't speak English. To make things worse, they weren't even told about the standards.

Oops.

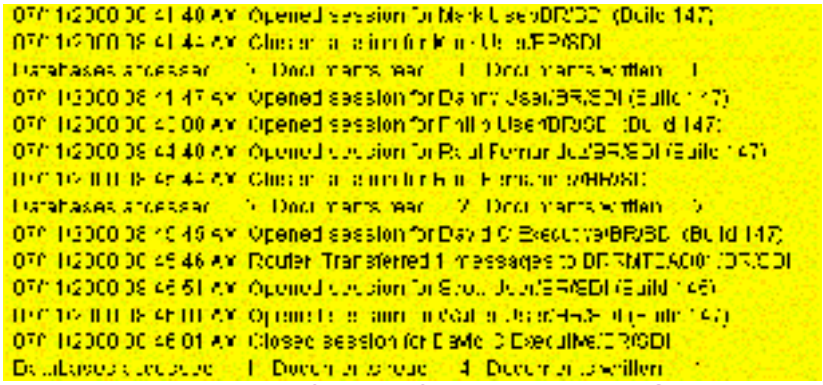
The one thing SDI did have, prep-wise, was a comprehensive list of the local administrators for every large office in the Americas. This list proved to be quite useful to me, after the fact.

Identification

Identifying the actual incident was easy. The probe was sanctioned ahead of time. I did it, and I kept good notes. However, I did my best to cover my tracks while the probe was in progress. I utilized one of the many Internet "privacy" proxy systems (like SafeWeb[5] or Anonymizer) for all of my web surfing. I waited a full day before making any approach at the server via Notes – and then plugged my laptop into a convenient jack at a local university. My Notes client was set to encrypt all network traffic. This was done to reduce the chance of anyone – good or bad – seeing what I was doing.

Little did I know at the time, but these simple actions essentially blinded the company's security systems to my actions. Since my traffic was "accepted" by the firewall, it logged only the IP address of the proxy I used and the number of the rule that caused me to be "accepted." Domino/web logging was not running on the target machine, so administrators had no way of telling which web surfer read what or when they did it. Going further, none of the more verbose logging options were in use at the Domino server level (see the Appendix at the end of this document for more information).

So, while it may have struck some people as odd that "Raul" connected to the Notes system at 8:45 AM (he worked the late shift), the administrators who later looked at the logs had no way of telling where he was when he did it. As you can see, there truly wasn't much to go on.



cleaned log.nsf excerpt from WebNotes/BR/SDI

The results were pretty dismal for passthru logging as well, which was off completely[6]. So, by all appearances, Raul Fernandez – or someone who copied his ID file – did a bunch of strange things right from the comfort of his office in Sao Paulo.

[5] Now in the process of evaluating a pay-only service. In the meantime, Safeweb.Com is no longer active.
[6] default setting for Domino

The other potential indicators – which included the modification time for the Passthru user group in the SDI-SA Address book – were so far removed from the earlier items; most people probably wouldn't make the connection. See the example below, which was created to mimic the original document's properties.

.My notes were the only solid evidence that something deeper happened. Sprinkled liberally with screen shots, I made sure they were thorough enough to allow anyone to virtually re-run the operation from the beginning at any time[7].

After completing my expedition, I seriously debated whether I should wait and see if anyone reacted to the intrusion, or if I should approach management immediately. Since my goal all along had been to highlight weaknesses so that they may be quickly repaired, I opted for the immediate approach. I reasoned that waiting would only prove that the local administrators weren't paying attention, and little else. Such a move would most likely just angered them, and gained me nothing.

I wrote a quick e-mail, informing two Domino administrators of exactly what I found (screenshot below). My boss ("Barney") was CC'd on the same message. "Anne" and "Vincent" were ostensibly the top administrative tier in the SDI Americas messaging deployment, and as such, held sway over their fellow admins in Brazil. Included in my original message were many of the screen shots shown in this document, a timeline narrative of the penetration, and a zip file containing the user ID files I cracked.


Karl Rademacher
07/11/2000 01:17 PM

To: Anne Translator/US/SDI@SDINA, Vincent Admin/US/SDI@SDINA
cc: Barney Boss/US/SDI@SDINA

Subject: Exposure via Rogue Web server in Brazil

*** THIS MESSAGE WAS SENT ENCRYPTED ***

Hi guys,

This message is being sent to inform you of a serious lapse in security involving a server in Brazil. Please read the attached word document concerning this machine, and respond with an intended course of action. The zip file contains all of the ID files which were copied during this exercise. I have destroyed the copies I had.

If possible, could you have the local administrators in Brazil respond to the enumerated points below? We need to get this cleaned up as quickly and quietly as possible.

1. There is apparently no firewall or IDS system protecting this machine. When will there be one in place?
2. Was authorization ever granted to have this server on the internet (from Messaging Management, Information Security or WAN Security)?
3. Now that we have it, who will harden this machine against malicious attack?
4. A web service shouldn't be running on a machine specifically used for an email gateway.
5. The ACL structure on this web server is such that it virtually invites unwanted attention. Anonymous access should not be allowed to sensitive data.
6. Leaving user and server ID files in the NAB is a dramatic fault in security.
7. Because the South America Address Book was compromised so easily, any other Address book on an SDI-SA server can be gotten as well. Domains based on the /SDI root certifier should be considered unsecure, at best - until we can determine depth of exposure and ascertain whether anyone else managed to do what I just did. (SDI-NA, SDI International, SDI-FE, SDI-EMEA, etc).

There are profound violations of our Network Security Policy, and they need to be fixed ASAP. The longer we wait to address this, longer we leave ourselves highly exposed to intrusion, or other malicious intentions. I eagerly await a response.

Karl R.

Point 3 in the document was necessary because it was apparent that several critical business functions had been built around this server by the time I got around to probing it. I knew there would be zero chance of ever

[7] Due to security reasons, a good portion of those notes are not included in this practical.

getting the plug pulled on this box. The SDI Information Security group had little or no visibility in South America, so our standards and such didn't get much consideration there.

Looking at the situation in hindsight, I think the administrators in Brazil were really stuck. On one side, there existed certain rules that they were marginally aware of – like informing the other messaging administrators of major changes they planned (a new SMTP gateway certainly qualified as major). On the other side, their bosses – the very people who actually signed their paychecks – probably demanded a local mail gateway AND the ability to check mail from home. In the process of putting voice to these demands, I'm sure the executives made it abundantly clear that any delays in implementation would have a negative impact on career longevity. This is theory, of course, but it lends credence to conclusions I'd reached about odd things I noticed both during and after the probe (like the ACL inconsistencies evident on pages 7 and 8 of this document).

At the time, had I known of resources like the FBI Law Enforcement Bulletin, I might have followed the lead of some of the nations leading criminal investigators. In FBI parlance, this incident neatly fit the description for a "Major Case" in that it cut cleanly across SDIs political fiefdoms, and the clean up eventually involved several subgroups within SDI. The incident had a long term impact on personnel, policy and resources. Also, had the results of my Whitehat exercise been made public (even within the confines of SDI), it would have attracted a great deal of attention – some from senior executive management.

I should have taken time to quickly determine not only the main players (outside Information Security) to be involved, but also each person's stake in what I was going to present, what role I expected them to take in the cleanup, their perception of the status quo, and how they'd react when I approached them. I should have – but didn't. As stated earlier, this was the first time for me handling an incident with this level of potential impact.

The immediate reaction to my message (and follow-up voice mail) was, quite understandably, defensiveness. No one likes being told that they were negligent – especially in a manner that could conceivably threaten the welfare of the entire corporation. No matter how you approach or sugarcoat an issue like this, somewhere deep down, the recipient of such news will interpret it as a personal attack. Better preparation on my part would have lessened this effect with Vincent, but probably not have eliminated it. To make matters worse, my initial tone was somewhat abrasive – and Vincent didn't react well to it. Recently and complete unknown to the Information Security group, Vincent was promoted to head of regional IT support for every SDI office south of Texas. He saw any public drubbing of an administrator under his umbrella as mud thrown at his personal reputation.

Before the containment process could even begin, I had to smooth ruffled feathers with Vincent. After all, I'd known him for four years and previously had gotten along well with him. System compromise or not, the last thing I wanted was Vincent angry with me. So, I called him. In the half-hour conversation that followed, I apologized for the tone of my initial letter, explaining that I was simply overwhelmed by the potential scope of impact

and unfortunately let it show in what I wrote. I continued by telling him that the root of the problem predated his tenure in management and the perception of Barney Boss and myself was that Vincent Admin inherited the problem; he didn't create it. Blame, if it could be assigned, lay squarely at the feet of the US messaging group – not the Brazilian one – for their almost willful neglect of their southern compatriots[8].

Now, a few strategic people outside of messaging knew Vincent's people weren't getting the support they needed, and would be working to correct that issue during the long term.

However, I continued, the laundry list of problems still existed and the onus was upon us, as a team, to fix them quickly. To make Vincent feel a little more secure, I informed him that not many people even knew of the penetration, and it was my intention to keep it that way. The whole operation was classified GRAVELY CONFIDENTIAL and he was one of the chosen few to know the full operational details of what actually happened. No one outside those immediately involved would ever find out, so long as the list of grievances was taken care of in a timely and efficient manner.

In the end, Vincent was mollified and promised to be a cornerstone resource in resolving this particular incident.

Containment

Within minutes of the end of our conversation, the web service running on WebNotes (a.k.a. mx.supdeduper.com.br) was permanently killed. By close of business the same day, the following corrective adjustments were also done:

- All User ID attachments were deleted from the SDI-SA Address Book.
- The Address Book was forced to pull changes from the hub server, resulting in the application of the more restrictive SDI standard ACL (examples on page 14):
    1. "Anonymous," "default access," "deny access" and "maximum internet access" all set to **no access.**
    2. "General users" was set to author.
    3. "Administrators" and "Hub servers" were both set to manager.
    4. "Spoke servers" was set to "editor."
    5. All groups were defined as the appropriate group type, and "Enforce Consistent ACL" was set.
- Every database on the server had "Anonymous" added to the ACL, with **no access** rights. In those databases where "Anonymous" existed before, the entry was changed to **no access**. The decision to do this was made easy by the fact that there wasn't a single authorized Domino Web server in the domain, thus making the need for anonymous access unnecessary. In addition to what was done to "Anonymous", both "Default Access" and "Maximum Internet Access" were set to "No Access."

[8] this assertion was helped by the fact that two managers in the US messaging group had recently left "involuntarily."

- Due to a related vulnerability, all copies of webadmin.nsf and webadmin.ntf on every server were sought out and deleted. We didn't need web access, so we most certainly didn't need web administration and the potential holes that it brought into play.
- The firewall rule for inbound web traffic (TCP Port 80) was edited to silently drop any web access to this server.
- The firewall rule which specifically allowed inbound Compaq Insight Manager (TCP port 2301) access to the SMTP box was deleted.
- The firewall rule which allowed Notes (TCP port 1352) connections from the DMZ to anywhere was deleted. A rule allowing Notes connections from the Intranet to the SMTP server in the DMZ replaced it.
- A Notes server connection was created in the address book which caused BRRMTSAO01/BR/SDI to connect to WebNotes/BR/SDI every ten minutes so they could exchange waiting mail, and BRRMTSAO01/BR/SDI could PUSH any database changes to WebNotes (any changes introduced on WebNotes were simply ignored).

The tool used for most of this (except the firewall) was Vincent's personal Notes client, located in Detroit. SSH was used for the Firewall.

Eradication & Recovery

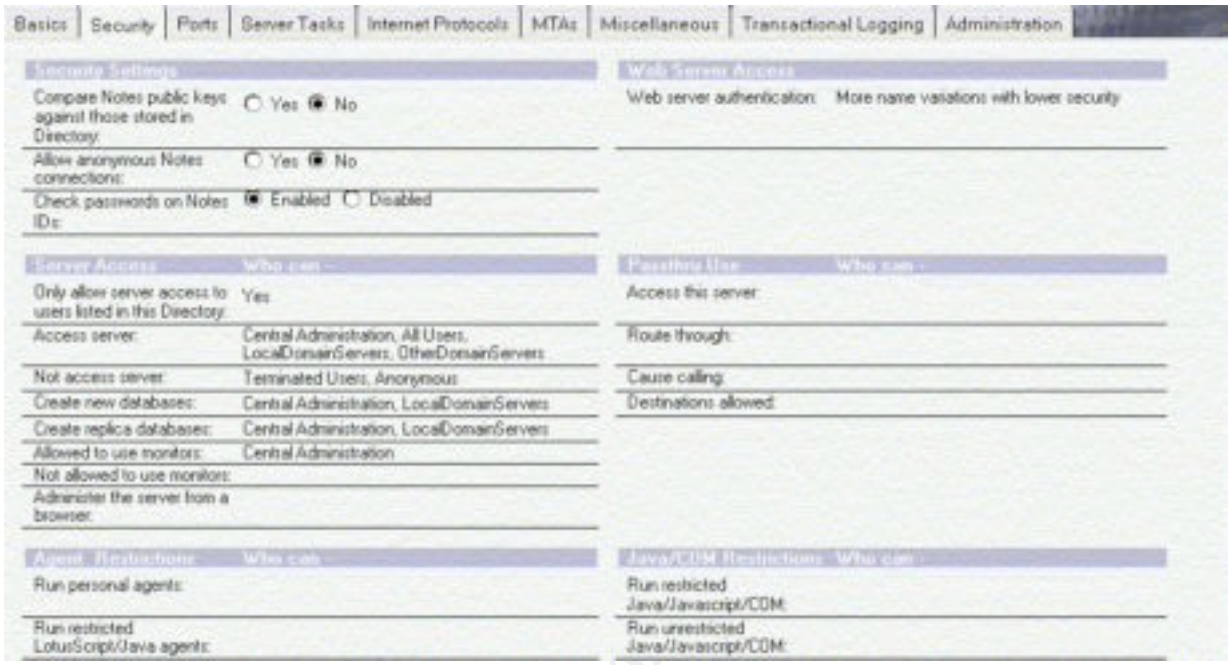
Because we couldn't guarantee that someone else hadn't done the same thing I did, a new Notes certifier ID (/Brazil/SDI) was generated the next day. Every user (400+) and server that had an ID file in the Address Book was forced to re-certify with it. Over the next week, the new ID files were hand delivered to each person, and old IDs were destroyed.

The Information Security team carefully pored through months of server logs for WebNotes, which we had restored from DLT. In these logs, we took note of database activity, agent executions, and user login / logout events, among other things. Even though we didn't find any overtly unusual activity, the Information Security team assigned to this incident still took note of every database which had a replica on WebNotes.

We then had trusted Domino developers inspect all design elements of each listed database as part of a "regular audit of Domino system security at SDI" [9]. In addition to looking for a checklist of security issues, we had them make design recommendations and had them check that all databases remained exactly as originally designed by whoever programmed them in the first place (whether a company employed programmer, and external consultant or Lotus itself). This was to make sure we didn't have programmers coming in after the fact and making undocumented alterations or patches to production applications – a problem that had become all-too-common lately. Where there were discrepancies, they were noted. However, beyond a few oversights and cut corners, nothing appeared to be out of the ordinary.

[9] The developers were not informed about the true reason for the database audits.

Concurrent to that effort, Vincent eliminated the passthru feature throughout the Domino domain. This was done by simply emptying all fields in the “Passthru Use” section of every server document in the address book (much like the example below).



Meanwhile, the real “Raul Fernandez” went to the smtp gateway, and performed the following actions:

- Removed Compaq Insight Manager completely.
- Updated the Operating System to NT 4.0 SP6A, with the latest security patches.
- Changed the passwords for all accounts at the operating system level, and implemented strict complexity requirements.
- Renamed the administrator account and deleted the guest account.
- Disabled all the services listed on page 11 of this document.
- Removed all shares.
- Performed a thorough virus scan.
- Dumped the NT server logs, and sent them to me. They appeared to be clean.

Since we couldn’t say for certain whether someone else had gotten into the machine before I did, Vincent made the unusual decision to completely replace it (unusual in the fact that the Brazilian office didn’t have a large budget for new equipment and rarely bought anything). Being fully aware of their fiscal restraints, Barney Boss arranged to have a recently retired server sent down to Sao Paulo at the Information Security group’s expense. While the machine in question no longer met the high-performance requirements of the main Detroit data center, it was more than powerful enough to handle what Brazil needed. Coordination of the delivery and install process took roughly two weeks.

The new server was built with a hardened version of Windows 2000, and ran the latest version of Domino (R5.04). It had a new Domino name, which conformed to SDI naming standards, and resided at a new IP address. The instant that the old machine was phased out, all references to it were removed from the address book.

Once this was done, the further step was taken to replace the copy of Names.Nsf on every server with a greatly sanitized version possessing a new replica ID. The new Name and Address Book (NAB) conformed exactly to the canned template provided by Lotus. This was done for the following reasons:

- If someone did pull a replica of the original database to an unknown location – using a new replica ID essentially cuts them off from getting any updates.
- If someone placed extra Lotus Script in a hard to find location in the NAB, it was now gone.
- If a view or form had been modified to provide extra information or NOT provide essential information, said modification was now gone.
- Sometimes administrators and developers take a short cut and add a new view or form to the NAB. While they may be benign, these additions can cause trouble occasionally. Such views or forms were now gone.
- Alterations to the NAB design invalidate any support Lotus will give for issues arising from the NAB.

To increase security, all servers had the “Compare Notes Public Keys” field set to “Yes,” and the “Web Server Authentication” field set to “Fewer Name Variations with Higher Security.” See the picture on the previous page to see where this is done.

All during this cleanup, Vincent and I were in constant contact. SDI purchased Internet Scanner from ISS. I took this and scanned the Brazilian DMZ three times in the week following the initial report. While I knew what to expect, I was hoping to NOT find anything else (new servers suddenly popping up, new services on old servers, previously unseen vulnerabilities, etc). This practice continues today, though on a monthly basis. I dutifully generate reports and send them to both my supervisor and to Vincent Admin.

An important occurrence within the last six months was the division of IT responsibilities in Central and South America. With the heightened focus on security and the need for better local expertise bolstering him, Vincent lobbied his business unit CFO into budgeting for more IT personnel. Previously, “Raul” and one other person divided up the entire office (350+ people) between them, and handled every kind of issue imaginable. Three more people were hired at the regional level to handle server administration duties, leaving “Raul” and the other technician free to handle user desktop issues, which they greatly preferred. The new people are all based in the U.S., are paid accordingly, and receive regular training.

As a result of all these steps, security issues with the South American region of SDI have been at a historic low.

Lessons Learned

People Skills

One person can't possible handle the clean up of every security event alone. Especially in a company the size of SDI, good people skills are a must! Even on the little incidents, I've learned to be a bit more circumspect in my initial approach to people. If I have the time, I try to learn a bit about someone beforehand. Most of the time, administrators, developers and users are unaware that a certain condition exists and are more than willing to do their part in cleaning things up – if asked. Managers and executives, by the same token, may not be aware of the impact to security that an edict or policy has. I go out of my way to maintain a good rapport with administrators, while the executives within SDI's security group do the same at their level. We all accompany this with constant, but gentle pressure. And guess what? It works.

Not only does a good rapport work toward breaking down the various barriers that may exist in the workplace, it helps you to assess the skills of those you may tap in the event of a large scale or remote incident.

Information Handling

Depending on the gravity of a situation, discretion should be exercised with regard to the dissemination of information. For example, while this particular incident was handled with a great deal of secrecy, others have been dealt with in a far more public manner. For two situations during the last year, SDIs Information Security group set up a war room, put into service an incident-specific web site, and activated a special 800 dial-in number. The Information Security group also tapped into known resources within human resources, the legal department, telecommunications and information technologies in order to make each of those incidents pass as quickly and as painlessly as possible. In general, we put on a very public appearance and it was well received.

At the quieter end of the scale, I've personally handled cases involving the dismissal of employees over the use of company assets to store and distribute inappropriate material (like pornography). Each case was a clear violation of SDI corporate policy. However, none of them were anything SDI wanted trumpeted from the top of the highest mountain. Neither are other cases where loose lips may result in business loss, reputation damage, or substantial liability for your employer.

To this end, SDI has a defined process by which each computer security incident is systematically reviewed by our CISO or one of our Regional Information Security Officers (RISO) and the designated Incident Owner within the Information Security group. This process is used to weigh the many variables which can influence how much knowledge of a specific incident is needed, and by whom.

Knowledge Gained & Proactive Behavior

My own appetite for Domino security knowledge increased dramatically as a result of this probe. Now, instead of just glossing over the various release notes (readme.nsf) included with each new version of Domino, I actually take the time to read them

thoroughly. I also keep tabs on security content in the Domino administrators help file (help5_admin.nsf). Since I stay in constant contact with the Messaging group at work, this helps me keep them abreast of vulnerabilities on servers they maintain.

All new databases or changes to existing databases at SDI must now undergo a formal security audit before being put into production. I am one of three people who have to approve these changes before they are allowed. Proof of how seriously this policy is taken by executive management came a few months ago, when a senior developer was dismissed after failing to abide by its guidelines.

A similar policy is in effect for any changes to the corporate networking infrastructure (routers, switches, servers, LAN/WAN segments, applications, etc).

With the recent reorganization of SDI's security group, I'm afforded a greater amount of time to dedicate to proactive tasks, such as researching new security tools, sandbox testing system compromises, and random probes of SDI computing assets.

Tuning in to my environment

The security group watches BugTraq and other similar venues closely. When a new vulnerability is announced, the first thing we do is try to do is determine its applicability to systems at SDI. With Domino and Windows announcements, I personally try to duplicate the exploit. I then forward the original announcement, any links to patches, and a summary of my findings to my supervisor AND to the Corporate Security Communications Officer. There are others in the security group who do the same thing with other operating systems and major packages.

Since the lion's share of our incidents involve virus outbreaks, we are all specifically trained to handle them – individually and as a group. In fact, SDI has codified several Global and Regional virus/worm incident response procedures, depending upon virulence and impact, among other variables.

The deepest lesson learned

In order to remain as free as possible from the effects of various security threats out there, we need to not only be prepared for the ones we know, but also constantly seeking the ones we don't know, and be able to react to all of them in an efficient and context-appropriate manner. Part and parcel to this is knowledge of the internal dynamics within our local organization and ultimately, SDI at the global level. Making that breathless announcement, and offering everything you can to help does no good if nobody is listening.

As Thomas Jefferson said, *"The price of liberty is eternal vigilance."*

References

Specific Publications and Articles

Subtle Skills for Building Rapport By Vincent A. Sandoval, M.A., and Susan H. Adams, M.A. FBI Law Enforcement Bulletin, August 2001

Major Case Management By Brian P. Carrol, M.S. FBI Law Enforcement Bulletin, June 2001

InfoWorld Security Advisor op ed article By Stuart McClure & Joel Scambray. January 5, 2001

Locking Down a Lotus Domino Server, by Andrew G. Hargreave, III. December 7, 2000

Lotus security advisory 189425, dated December 3, 2001

Lotus's Guide to Secure Domino Applications, dated November 11, 2001

The Notes,Net Special Security Issue of Iris Today, dated September, 2001

Lotus Notes and Domino R5.0 Security Infrastructure Revealed. Fiona Collins, lead author. May 6, 1999. ISBN 0738413089

Lotus Domino View ACL Bypass Vulnerability Listing at Security Focus, October 30th, 2001

Hack Proofing Lotus Domino Web Server by David Litchfield. 21st October 2001

General guidance and other resources

The Lotus Corporate Website

The Lotus Notes.Net website

Foundstone Security Services

The Computer Security Institute

BugTraq

Dominosecurity.org

SANS

Security Focus

2600, The Hacker's Quarterly

Attrition

Safemode

@Stake

CERT.ORG

Tools

Eeye.Com Tools

Sam Spade

Snort.Org

Arne Vidstrom's Toolbox

Next Generation Security Software

The SANS Internet Storm Center

SafeWeb

Anonymizer

Network Ice

ISS X-Force

Appendix: Some helpful Notes.INI log settings

Note: Except for minor formatting changes, the information listed in this appendix is copied directly from the **Domino 5 Administrators Help** database for Version 5.08 of Lotus Domino. The default location for this database is help\help5_admin.nsf on PCs where the Administrators Client has been installed. The information – and more – can be accessed opening this database and clicking "index," then clicking on "Notes.INI settings."

Log

Syntax: Log=*logfilename, log_option, not_used, days, size*
Description: Specifies the contents of the log file and controls other logging actions:

Parameter	Value
logfilename	The log database file name, usually LOG.NSF
log_option	Log options: 1 = Log to the console 2 = Force database fixup when opening the log file 4 = Full document scan
not_used	Always set to zero; this parameter is not currently used
days	The number of days to retain log documents
size	The size of log text in event documents

For example: Log=LOG.NSF,1,0,7,20000
The log file (LOG.NSF) is deleted in seven days and can contain up to 20,000 words. All log information is also sent to the console.
Applies to: Servers
Default: Log=LOG.NSF,1,0,7,40000
UI equivalent: None

Log_AgentManager

Syntax: Log_AgentManager=*value*
Description: Specifies whether or not the start of agent execution is recorded in the log file and shown on the server console:

0 – Do not log agent execution events
1 – Log agent execution events (partially and completely successful)
2 – Log agent execution events (completely successful only)

Applies to: Servers
Default: None
UI equivalent: None, although you can specify this setting in the NOTES.INI Settings tab of the Configuration Settings document in the Domino Directory.

Log_DirCat

Syntax: Log_DirCat=1
Description: Logs the following information about the Directory Cataloger task to the Miscellaneous Events view of the log file (LOG.NSF):

When the Directory Cataloger starts
Which directories the Directory Cataloger runs on
When the Directory Cataloger finishes

Applies to: Servers
Default: None, although without this setting the log file only shows when the Directory Cataloger starts.
UI equivalent: None

Log_Replication

Syntax: Log_Replication=*value*
Description: Specifies the level of logging of replication events performed by the current server:

0 – Do not log replication events
1 – Log that a database is replicating
2 – Log summary information about each database
3 – Log information about each replicated document (both design and data documents)
4 – Log information about each replicated field

Applies to: Servers
Default: None
UI equivalent: None, although you can specify this setting in the NOTES.INI Settings tab of the Configuration Settings document in the Domino Directory.

Log_Sessions

Syntax: Log_Sessions=*value*
Description: Specifies whether individual sessions are recorded in the log file and displayed on the console:

0 – Do not log individual sessions
1 – Log individual sessions

Applies to: Servers
Default: None
UI equivalent: The Log All Client Events setting that is an Advanced server Setup option. You can also specify this setting in the NOTES.INI Settings tab of the Configuration Settings document in the Domino Directory.

Log_Tasks

Syntax: Log_Tasks=*value*
Description: Specifies whether the current status of server tasks is recorded in the log file and displayed on the console:

0 – Do not send status information
1 – Send the status of server tasks to the log file and to the console

Applies to: Servers
Default: None
UI equivalent: None, although you can specify this setting in the NOTES.INI Settings tab of the Configuration Settings document in the Domino Directory.

Log_Update

Syntax: Log_Update=*value*
Description: Specifies the level of detail of Indexer events displayed at the server console and in the log file:

0 – Records when the Indexer starts and shuts down.
1 – Records when the Indexer starts and shuts down and when the Indexer updates views and full text indexes for specific databases.
2 – Records when the Indexer starts and shuts down and when the Indexer updates views and full text indexes for specific databases. Also records the names of views the Indexer is updating.

Applies to: Servers
Default: None
UI equivalent: None

Log_View_Events

Syntax: Log_View_Events=*value*
Description: Specifies whether messages generated when views are rebuilt are recorded in the log file:

0 – Do not log messages when views are rebuilt
1 – Log messages when views are rebuilt

Removing this setting from the NOTES.INI file also disables logging of these messages.
Applies to: Servers
Default: None
UI equivalent: None

Mail_Log_To_MiscEvents

Syntax: Mail_Log_To_MiscEvents=*value*
Description: Determines whether all mail event messages are displayed in the Miscellaneous Events view of the log file:

0 – Does not display mail events in the Miscellaneous Events view
1 – Displays mail events in the Miscellaneous Events view

Applies to: Workstations and servers
Default: None, although if this setting is omitted, mail events are not displayed in the Micellaneous Events view
UI equivalent: None

No_Force_Activity_Logging

Syntax: No_Force_Activity_Logging=*value*
Description: Controls whether the Statlog task automatically enables activity logging on all databases:

0 – Allows automatic activity logging on all databases
1 – Prevents automatic activity logging on all databases

Even when activity is not being recorded for the database, the information is still recorded in the Activity entry of the Database Usage view in the server's log file.
Applies to: Servers
Default: None, although if this setting is omitted, the Statlog server task enables the Record Activity option for every database on the server and adds 64KB to each database.
UI equivalent: None, although you can specify this setting in the NOTES.INI Settings tab of the Configuration Settings document in the Domino Directory.

Passthru_LogLevel

Syntax: Passthru_LogLevel=*value*
Description: Specifies the level of trace information recorded for all network connections (including passthru) in the Miscellaneous Events view of the log file.

0 – No information is recorded
1 – Only errors are recorded
2 – Summary progress information is recorded
3 – Detailed progress information is recorded
4 – Full trace information is recorded
5 – Full trace information plus driver messages are recorded

Applies to: Workstations and servers
Default: 0
UI equivalent: File – Preferences – Notes Preferences – Ports – Trace – Notes Log options

PhoneLog

Syntax: PhoneLog=*value*
Description: Specifies whether phone calls are recorded in the log file:

0 – Does not record phone calls to the log file
1 – Records all calls, except those that fail because of a busy signal
2 – Records all phone calls

Applies to: Workstations and servers
Default: 2
UI equivalent: None, although you can specify this setting in the NOTES.INI Settings tab of the Configuration Settings document in the Domino Directory.

RTR_Logging

Syntax: RTR_Logging=*value*
Description: Enables or disables monitoring of Cluster Replicator activity.
Use the following values to set this variable:

0 – Disables monitoring of the Cluster Replicator
1 – Enables monitoring of the Cluster Replicator

Applies to: Cluster servers
Default: None
UI equivalent: None

Server_Show_Performance

Syntax: Server_Show_Performance=*value*
Description: Specifies whether or not server performance events are displayed on the console.

0 – Records server performance events in the log file
1 – Displays server performance events on console

Applies to: Servers
Default: None, although if this setting is omitted, Domino records server performance events in the log file
UI equivalent: None, although you can specify this setting in the NOTES.INI Settings tab of the Configuration Settings document in the Domino Directory.

SMTPDebug

Syntax: SMTPDebug=*value*
Description: Controls the level of console logging performed by the SMTP task.

0 – No logging
1 – Log errors
2 – Log Protocol commands

Applies to: SMTP servers
Default: 0
UI equivalent: None

SMTPSaveImportErrors

Syntax: SMTPSaveImportErrors = *value*
Description: Specifies whether mail message import errors are recorded, as follows:

0 – No messages are recorded.
1 – When an arriving message fails to be written as a note in MAIL.BOX, Domino writes the data stream to a temporary directory, and logs the name of the file.
2 – All arriving messages have their data streams written to the temporary directory.

Note: This feature can use a great deal of disk space since the saved messages continue to accumulate until you delete them. Also, the content of the messages are accessible to anyone with the privileges to read files in the temporary directory.
Applies to: Servers
Default: 0
UI equivalent: None

WebAuth_Verbose_Trace

Syntax: WebAuth_Verbose_Trace=1
Description: Enables a Domino Web server to record at the server console detailed information about specific Web user authentication sessions, for example: authentication success or failure; the group cache information used to verify Web users' membership in groups for database access control; the search filters used to find user and group entries in an LDAP directory.
Use this setting to troubleshoot problems with Web server user authentication and Web server group searches for database access verification. After you correct the problem, make sure to disable this setting -- remove it or set it to 0 -- because using it slows Web server performance.
Applies to: Web servers
Default: None
UI equivalent: None