Global Information Assurance Certification Paper

SANS/GIAC GCIH Practical Assignment

Windows system compromise Incident through a Unicode exploit.

By Potheri Mohan, Loudcloud, Inc.

Executive Summary:

On May 8, 2001 a remote attack was launched on a company managed IIS web server. The attack on the Windows 2000 server running IIS was successful. As a result the server was compromised and the site was defaced. The original web page was replaced by anti US government propaganda.

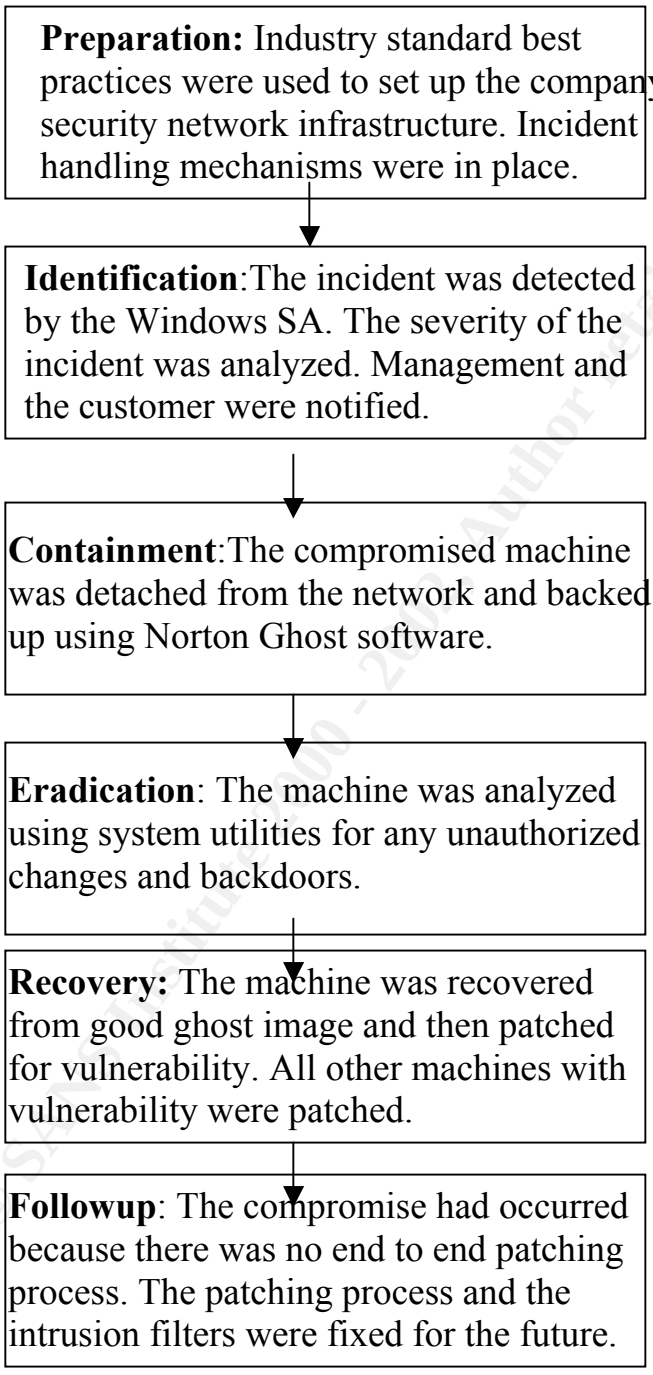
The compromise was detected by the systems administrator for the site, two days after the incident had happened. Since the machine compromised was used only occasionally and was not the primary web server for the customer, the effect of the compromise. On learning of the compromise, the attack was analyzed and was found to have a severity of three.

The security incidence response team was called into action. Investigation showed that only one of the two machines used redundantly for the web site was compromised. The compromised machine was shut down, with the second machine taking over during the recovery phase. The machine was fully recovered using a good image and was patched prior to being brought back on line. The second machine was also patched for the vulnerability.

Summary of Activity

- Machine compromise was detected
- Management and customer was contacted.
- Firewall and Snort IDS logs were analyzed for the incident
- Severity was calculated for the incident.
- Compromised machine was taken off the production network.
- A ghost image of the compromise machine was created and stored.
- The machine was analyzed using Windows and Resource kit utilities
- The machine was restored from good backup image.
- Machine was patched for vulnerability and brought back on line
- Details about the attacker were obtained from whois and stored as evidence.

Steps in the Incidence Response:



Phase 1:Preparation

The preparation phase for the organization are all the steps that it has taken to ensure a good security infrastructure. The company has created several security policies that encompass different areas of security.

The company has established the following security best practices:

Vulnerability Assessment:

Nessus based Vulnerability assessment of all exposed hosts for the latest known vulnerabilities. All hosts exposed to the Internet are audited before they first go live and after that they are audited on a regular basis. The vulnerability database is updated regularly for the latest known vulnerabilities

Intrusion Detection:

Network based Intrusion detection using Snort at all Internet connected entry points. Snort IDS sensors are placed behind the border routers in every data center and can see all the network packets going to and from the company hosted network and the Internet. Snort filters are update regularly for new attacks and vulnerabilities. All snort alerts are passed via syslog to the security monitoring organization and all genuine alerts are looked at alerted on.

Host based Intrusion detection is implemented through various mechanisms. All critical system binaries are check summed and regularly checked for any kinds of modifications. All activities on the system are logged to a central syslog host in the data center. The information is then passed on to security monitoring, where filters are used to identify suspicious and attack activity.

Access Control:

All administrative access to the data center servers and networks are on a need only basis. All connections from outside the data center are allowed only through encrypted mechanisms. All connections have to go through only bastion hosts that are choke points for entry into the data centers. Securid based one time password authentication is used for logging into the data centers. Access to individual machines in the data centers are controlled granularly and should conform to the Network access policy.

External Port Scans:

Special hosts located on the Internet scan the public address space of the company with a subset of ports on a regular basis to get a good picture of what ports are being seen as open the Internet. This is then compared to the approved ports for each of the hosts and any compliance issues with company security policies are detected and corrected, in addition to the firewall rule sets being ascertained.

Security Audits:

Extensive security audits are conducted for all network equipment and servers prior to them going alive. The hosts are checked for all possible security issues such as access control, file permissions, patches, vulnerabilities, etc. The firewall rules are audited for conformance with company security policy and to ensure only approved ports, hosts and applications are allowed through. Audits are also done on a regular after go live or when a new host is added to the network.

Patch monitoring and Certification:

The company personnel monitor vendor and security mailing lists for the latest developments, vulnerabilities and patches released. When a new patch is released the patches are tested in a staging environment, before being certified for production. On completion of the certification all pertinent systems administrators and other engineers are notified about the vulnerability and the patch.

Server Hardening:

All servers are built from different system images that are hardened using industry standard checklists and guidelines. The images are constantly updated with new security patches as and when they become available.

Security Policies:

The company has good security policies with strong commitment from upper management. Policies impose mechanisms such as warning banners for all types of logins, non disclosure agreements, secure communication channels, etc. Security awareness in the organization is brought about beginning at employee orientation, at company business seminars and through publicity of latest attacks and attack trends.

Incidence Response readiness:

The company has created an incidence response team to deal with all types of security incidents. There is an incidence response plan that calls for involvement of members of the network operations center, systems administration, Information security, network engineering and legal departments. The plan provides for the following:

- Classification of different kinds of incidents and the corresponding action to be taken
- Tools for evidence collection and storage.
- Contact numbers for the incidence response team.
- War room for incident analysis.
- Out of bound communication channels via pagers and private IRC chat.
- Private backup network with backup software and devices.

Phase 2: Identification

In this phase, it is to be determined if an incident has occurred, its nature and impact on business. The lead NT systems administrator working on some routine activities on one of the Internet facing servers found anomalies in an IIS server running on Windows 2000. The home page of the machine had been defaced with an anti government message.

The incidence response team convened and discussed the tasks that needed to get done based on the incidence response plan. The security manager was designated as the primary person who would be managing the incident. A remedy ticket was opened was opened with the company's network operations center (NOC) and assigned to the on call engineer for the particular machine. The customer whose site was affected was notified about the incident and briefed about the incidence response that would be taken.

Since this is a web site defacement, all public traffic to that site was directed to a maintenance page. The on call applications engineer, the security engineer and the systems administrator were paged. The NOC was told not to do anything with the particular network that was compromised until further notice. Other hosts on the compromised network were systematically checked for any evidence of compromise. The legal department was consulted and their help was sought in evidence collection.

The security engineer and the systems administrator worked together to ascertain the genuineness of the incident. They went through the Windows event log to determine the time and date of the occurrence and to collect other evidence. The Intrusion detection system was also looked at determine, if the attack was seen by the system. The security monitoring organization was also contacted to obtain information on the attack and compromise. They were able to duplicate and confirm the data obtained from the Network based Intrusion detection systems. An incident log book was used to take notes during the entire incident handling process.

Looking at the Intrusion detection logs over several days, there were no real reconnaissance that was done from the attacking host prior to the actual attack. It seems like this was an instance of the sadmind worm that attacked Solaris boxes and the compromised boxes were then used to attack machines having the IIS Unicode vulnerability.

The following contains details of the attack taken out of the organization's logbook.

Incident Type:

Remote compromise and site defacement based on the IIS Unicode exploit.

Incident Location:

Machine was a Windows 2000 machine located on the company's east coast data center. Machine is dedicated to supporting a particular customer and was located behind a load balancer (VIP). Only one machine in a pool of machine for that particular VIP was found compromised.

Incident detected by

Incident was detected by the systems administrator for the Windows 2000 server, while performing routine systems administration activities. The home page for the IIS server had been replaced with anti US government content and seemed to indicate ownership to

Severity:

An analysis for severity of the incident indicated the following.

Severity = (Criticality + Lethality) – (System + Net) = (4 + 5) – (2 + 4) = 3

Criticality = 4	Machine was customer web server in production. But it was not the main web site, but an ancillary web site, that is only reached from other servers of the customer and is not frequently accessed.
Lethality = 5	The attack resulted in the host being compromised and the site to be defaced.
System=2	The operating system was Microsoft Windows 2000, which is relatively new. The problem was that the machine was not patched for the latest IIS vulnerabilities almost three months after Microsoft released its patches.

Net = 4	Access to only port 80 the IIS web server port was available from the outside. The router and the firewall were found to be configured securely with the principle of least privilege. Authentication was performed by secure password and access was also allowed only after logging into a bastion with Securid keyfob based authentication.

Probability of the source address was spoofed :

The nature of the attack requires two way communication and hence the attack could not have been spoofed. In addition, Ingress and Egress filtering were in place on the perimeters to prevent spoofing of RFC1918 addresses.

Evidence of Active targeting:

From the data, it was evident that the attack was spawned by a worm, which was attacking everything on the network that it was finding vulnerable. Attack from the same machine on other company IP addresses in different data centers with IIS running seemed to indicate that though a particular vulnerability was being used, no particular customer was being targeted.

Correlation:

Posting on Bugtraq " Another hacked unicode Box" May 2001
http://www.securityfocus.com/frames/?content=/vdb/bottom.html%3Fvid%3D1806

Analysis of the alert:

The logs for the attack were obtained from the Cisco Pix firewalls, which was logging to a Linux server running Syslog-ng.

May 8 04:58:50 172.16.73.148 May 07 2001 22:06:10: %PIX-5-304001: 63.141.3.20 A
ccessed URL X.X.64.170:/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir
May 8 04:58:51 172.16.73.148 May 07 2001 22:06:10: %PIX-5-304001: 63.141.3.20 A
ccessed URL X.X.64.170:/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir+..\
May 8 04:58:51 172.16.73.148 May 07 2001 22:06:10: %PIX-5-304001: 63.141.3.20 A
ccessed URL X.X.64.170:/scripts/..%c0%af../winnt/system32/cmd.exe?/c+copy+\win
nt\system32\cmd.exe+root.exe

May 8 04:58:52 172.16.73.148 May 07 2001 22:06:11: %PIX-5-304001: 63.141.3.20 A
ccessed URL X.X.64.170:/scripts/..%c0%af../winnt/system32/cmd.exe?/c+copy+\win
nt\system32\cmd.exe+root.exe
May 8 04:58:54 172.16.73.148 May 07 2001 22:06:13: %PIX-5-304001: 63.141.3.20 A
ccessed URL X.X.64.170:/scripts/..%c0%af../winnt/system32/cmd.exe?/c+copy+\win
nt\system32\cmd.exe+root.exe

Description of Log output from PIX firewall:

May 8 04:58:50 172.16.73.148 May 07 2001 22:06:10: %PIX-5-304001: 63.141.3.20
Accessed URL X.X.64.170:/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir

- The first field is the time stamp for the event.
- The second field is the IP address of the firewall device.
- The third field gives PIX message number and the type of message.
- The fourth field gives details of the URL that is accessed.

From looking at the CISCO PIX firewall logs above , we see that the attacker is accessing the unauthorized script directories on the server through the Unicode exploit. The user is able to execute cmd.exe and can perform file copying and other activities.

Data from the Snort:

The snort intrusion detection sensor also detected the alert through the plugin for http decoding. Attached is a hex dump of the attack packet and the Snort output

[**] spp_http_decode: IIS Unicode attack detected [**]
0508-04:58:53.477113 63.141.3.20:7850 -> x.x.x.20:80

This is output from the snort http decode plugin has the following fields:

- The first field is the plug in reference.
- The second field is the type of attack.
- The third field gives the date and time stamp
- The fourth field is the IP address and port of the attacking host
- The fifth field is direction of the connection
- The sixth field is the destination IP and the port number.

Below is a hex dump of the attack packet and the Snort output

TCP TTL:255 TOS:0x0 ID:0 IpLen:20 DgmLen:481
AP Seq: 0x0 Ack: 0x0 Win: 0x0 TcpLen: 20
14 40 68 CD 14 40 2C 30 30 39 2C 39 38 34 0D 0A .@h..@,009,984..
62 79 74 65 73 20 66 72 65 65 26 6E 62 73 70 3B bytes free

```
3C 2F 66 6F 6E 74 3E 3C 2F 74 74 3E 3C 2F 62 3E   </font></tt></b>
3C 2F 64 69 72 3E 0D 0A 3C 2F 64 69 72 3E 0D 0A   </dir>..</dir>..
3C 2F 64 69 72 3E 0D 0A 3C 2F 64 69 72 3E 0D 0A   </dir>..</dir>..
3C 2F 74 64 3E 0D 0A 3C 2F 74 72 3E 0D 0A 3C 2F   </td>..</tr>..</
74 61 62 6C 65 3E 0D 0A 0D 0A 3C 74 61 62 6C 65   table>....<table
20 42 4F 52 44 45 52 3D 30 20 43 45 4C 4C 53 50    BORDER=0 CELLSP
41 43 49 4E 47 3D 30 20 43 45 4C 4C 50 41 44 44   ACING=0 CELLPADD
49 4E 47 3D 30 20 57 49 44 54 48 3D 22 31 30 30   ING=0 WIDTH="100
25 22 20 42 47 43 4F 4C 4F 52 3D 22 23 46 46 46   %" BGCOLOR="#FFF
46 30 30 22 20 3E 0D 0A 3C 63 61 70 74 69 6F 6E   F00" >..<caption
3E 3C 54 42 4F 44 59 3E 0D 0A 3C 62 72 3E 3C 2F   ><TBODY>..<br></
54 42 4F 44 59 3E 3C 2F 63 61 70 74 69 6F 6E 3E   TBODY></caption>
0D 0A 0D 0A 3C 74 72 3E 0D 0A 3C 74 64 20 57 49   ....<tr>..<td WI
44 54 48 3D 22 31 30 30 25 22 3E 3C 62 3E 56 75   DTH="100%"><b>Vu
6C 6E 65 72 61 62 6C 65 20 74 6F 20 55 6E 69 63   lnerable to Unic
6F 64 65 20 23 31 33 3A 26 6E 62 73 70 3B 3C 2F   ode #13: </
62 3E 0D 0A 3C 62 72 3E 3C 62 3E 68 74 74 70 3A    b>..<br><b>http:
2F 2F 78 78 2E 78 78 2E 78 78 2E 78 78 2F 6D 73    //xx.xx.xx.xx/ms
61 64 63 2F 2E 2E 5C 25 65 30 5C 25 38 30 5C 25   adc/..\%e0\%80\%
61 66 2E 2E 2F 2E 2E 5C 25 65 30 5C 25 38 30 5C   af../..\%e0\%80\
25 61 66 2E 2E 2F 2E 2E 5C 25 65 30 5C 25 38 30   %af../..\%e0\%80
5C 25 61 66 2E 2E 2F 77 69 6E 6E 74 2F 73 79 73   \%af../winnt/sys
74 65 6D 33 32 2F 63 6D 64 2E 65 78 65 5C 3F 2F   tem32/cmd.exe\?/
63 5C 2B 64 69 72 2B 63 3A 3C 2F 62 3E 3C 2F 74   c\+dir+c:</b></t
64 3E 0D 0A 3C 2F 74 72 3E 0D 0A 3C 2F 74 61 62   d>..</tr>..</tab
6C 65 3E 0D 0A 0D 0A 2D 41                        le>....-A
```

Looking at this trace, we see the injection of Unicode characters in the URL to traverse backwards on the directory tree and to execute the cmd.exe program.

The exploit:

Overview

This is the new sadmin/IIS worm, it spreads using rcp, through vulnerable sadmin hosts. It also scans for vulnerable IIS boxes, which it then proceeds to deface. Made up of sadmin-brute, grabbb, and a couple of perl scripts. It leaves the bindshell (from the sadmin exploitation) open on 800/tcp, and also (for propagation purposes) adds '+ +' to ~root/.rhosts. The second phase is to use a vulnerability that exists in Microsoft IIS 4 and 5 such that an attacker visiting an IIS web site can execute arbitrary code with the privileges of the IUSR account. This vulnerability is known as the "Web Server Folder Directory Traversal" vulnerability.

Microsoft IIS 4.0 and 5.0 can be vulnerable to double dot "../" directory traversal exploitation. Remote unauthenticated users can access any known file in the context of the IUSR account. Under normal circumstances, attempts to access a file in this manner will fail. Any attempt to execute a file in a non executable directory will be rejected under normal circumstances. There is a set of default directories in the web folder that come with IIS; that includes the *scripts* directory, which is executable by default. Under normal circumstances only scripts in this directory and other specifically authorized directories can be executed on the server.

Under normal circumstances an access like this

http://www.example.org/data/../../../winnt/program.exe

will not download or execute the program "program.exe". If an attacker encodes the relative reference part of the URL in Unicode, the protection mechanisms in IIS fail and the program is executed. This bug in IIS with respect to Unicode decoding can be used to run unauthorized commands and to compromise the server.

Phase 3: Containment

Containment is used to limit the scope or magnitude of an incident. In this situation, we had to make sure that the same mechanism was not used to attack other hosts belonging to the same network. Web traffic is allowed through the firewall and then it hits a load balancer that listens for the IP address of the web server, which is actually a virtual IP (VIP). The load balancer is used to balance the load across two machines in this instance and the web traffic is sent to one of the two machines, which have their own private IP addresses. When the Unicode attack came across, it was directed to one of the two hosts and the host that was chosen was defaced, while its redundant partner was unscathed. There was one other host that shared the function of the current host, and it was to be ensured that this host was not compromised.

Securing the subnet:

Special monitoring mechanisms were put in place to look for all Unicode attacks that were coming to the hosts represented by the VIP. After verification that the second and redundant machine for the network was not compromised, the compromised machine was taken offline.

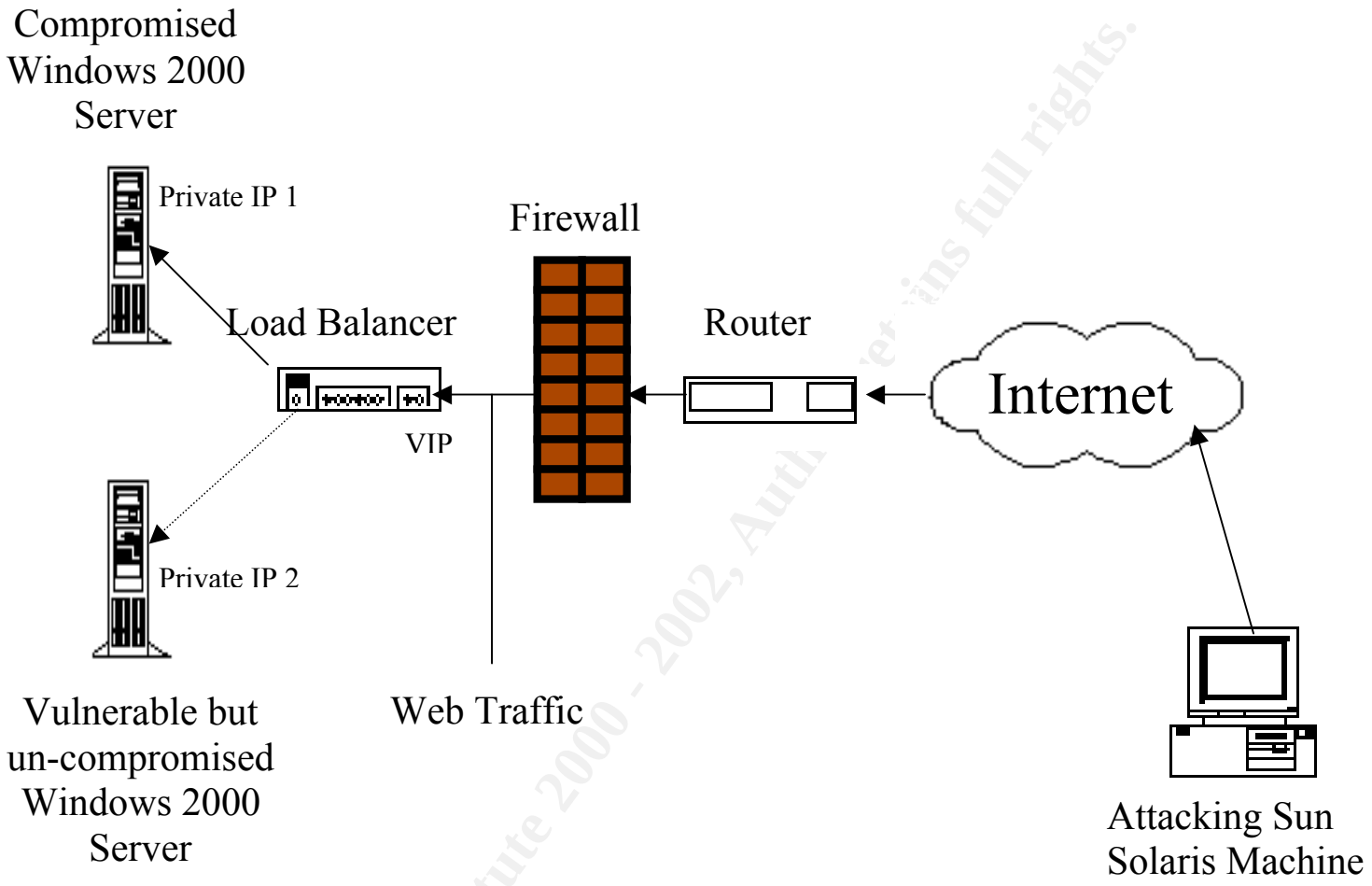


Figure 1: Network Schematic

Backup of compromised machine:

The incidence response plan calls for an image backup of the compromised machine. The machine was detached from the load balancer and a ghost image of the compromised machine was transferred and stored in the imaging server. Norton Ghost was used to create an image of the hard drives on the machine.

The ghost software has the capability to clone entire file systems for most commonly used Windows and Linux filesystems. The main purpose of the backup was to preserve the state of the machine and the evidence for further investigation. See the reference section for the link on ghosting.

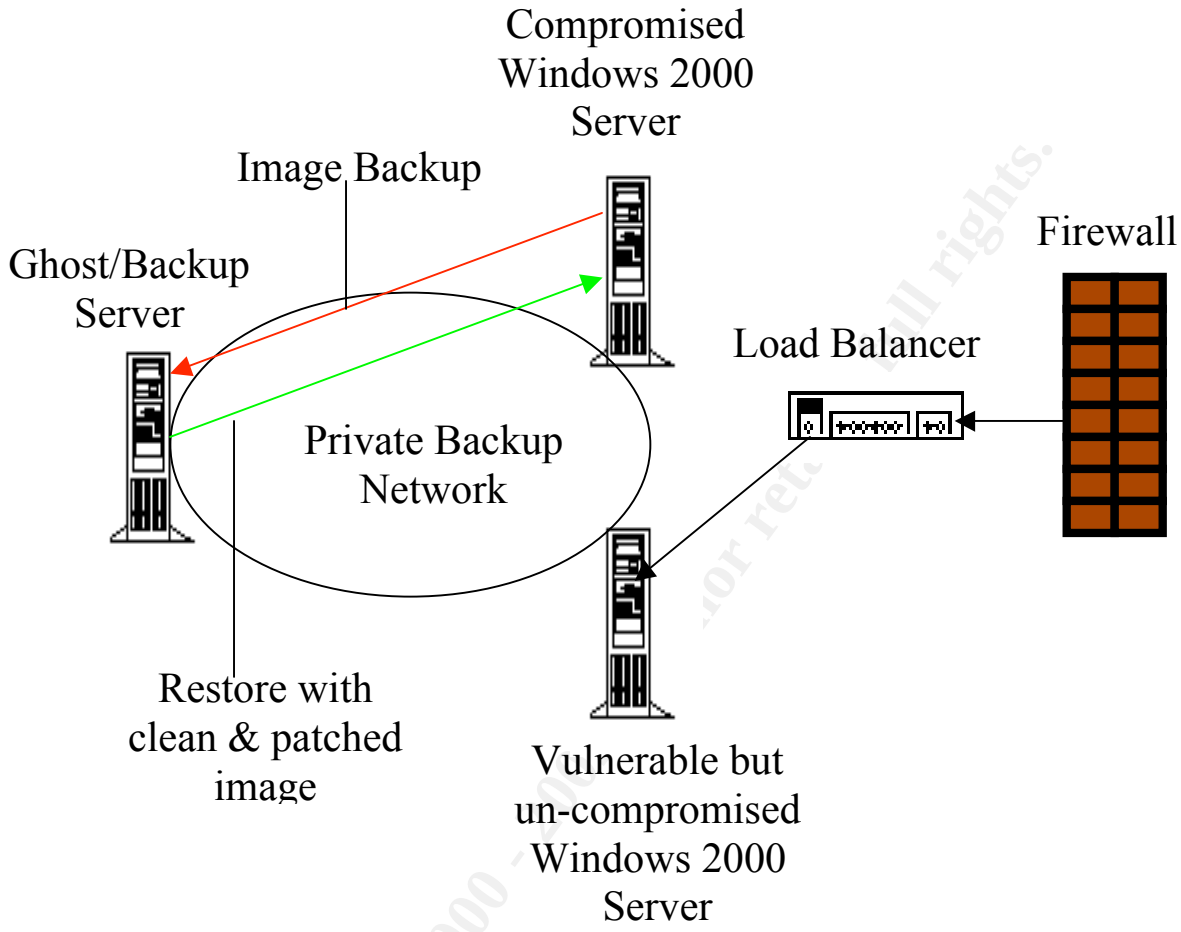


Figure 2: Backup Restore Mechanism

Phase 4: Eradication:

The purpose of the eradication phase is to remove the cause of the incident. In addition to undoing the damage that has been done, the cause of the problem should be determined. The incident handling guidelines were followed to perform the following eradication activities.

Activity:
The event logs and all the other logs on the system were checked for any unusual activity.

Findings:

Except for the actual Unicode attack and the directory traversal seen in the IIS logs, nothing unusual was seen in the logs

Activity:
The system binaries were checked against their checksums stored in the central image server.

Commands:
The sysdiff command was used to check for changes

```
sysdiff /diff original.img changes.img
```

The original.img file contains the original snapshot of the system. The first command compares the original snapshot with current snapshot and creates a diff image, which is changes.img

```
sysdiff /dump changes.img changes.txt
```

The above command is run to output the changes in human readable form.

Findings:
The system binaries and their checksums were intact.

Activity:
The system was checked for any kind of packet sniffers or netcat listeners.

The commands netstat and fc were used to check the ports on the machine.
An initial baseline had been created with

```
netstat –nav  >  baseline-ports.txt
```

The current port list was generated using

```
netstat –nav  >  current-ports.txt
```

fc was used to compare the current ports against the original ports.

```
fc –N baseline-ports.txt current-ports.txt
```

Findings:
No new ports were found on the machine.

Activity:
The users and groups were checked for any kind of modifications.

The commands addusers and fc were used to check the ports on the machine.
An initial baseline had been created with

```
addusers \\XXXXX /d baseline_users.txt
```

The current user list was generated using

```
addusers \\XXXXX /d current_users.txt
```

fc was used to compare the current ports against the original ports.

```
fc –N baseline-users.txt current-users.txt
```

Findings:
No new users or groups were found on the machine.

Activity:
The scheduler service was checked to see, if anything has changed. The "at" command was used for this.

Findings:
Nothing suspicious was found.

Activity:
The system was searched thoroughly for hidden files.

Findings:
None were found

Activity:
The system and network settings were checked for any changes. The registry was dumped using the regdmp utility and compared to the baseline registry.

```
regdmp –m \\XXXXX /d baseline_reg.txt
```

The current registry was generated using

```
regdmp –m \\XXXXX /d current_reg.txt
```

fc was used to compare the current ports against the original ports.

```
fc –N baseline-reg.txt current-reg.txt
```

Findings:
No changes were found

Phase 5: Recovery

The recovery phase is where the system is brought back into normal operational mode. A decision needs to be made in this phase about when the system is ready to go back into operation.

Since there is a redundant machine with identical data without being compromised, the task of recovery was made easy. The compromised machine was taken offline and rebuilt. Figure 2 shown earlier has the backup diagram. Norton Ghost was used to rebuild the machine from its binary image that was certified to be good. The Security patch provided by Microsoft based on the advisory MS01-026 was used to patch the machine. This is a cumulative patch that patched the IIS Unicode vulnerability and the latest security advisories from Microsoft.

After all the files were recovered and the machine was patched, the system was brought back on line and became part of the redundant server configuration with the load balancer. The backup machine that was serving the web site, when the compromised machine was being recovered was then brought down and patched for the vulnerability before being brought back on line.

The exploit was then run on both the machines to verify the absence of the vulnerability. The patch had closed the vulnerability and the exploit proved unsuccessful. The details about the original attacker were collected from the whois database and documented for future investigation.

Phase 6: Follow up and Lessons Learned

The incident was a big eye opener as far as security awareness for the company. Until then Security patches were not taken seriously and it was not the priority of the systems engineers to apply security patches. The compromise of the system brought a sense of urgency and importance to security.

The main reason the exploit had happened was due to the absence of an end to end security patch process. New vulnerabilities were identified by the security team and certified by the operations certification team, but there was no process in place to ensure that the patches were being applied.

A task force was setup to look into the patch process and to develop an end to end patching solution for the company. The following week two more machines having the same version of IIS were attacked, but because all machines with this vulnerability were patched, the attacks did not succeed.

The Intrusion detection system with snort was primarily used as a daily reporting mechanism rather than for real time alerting. After this incident, the rule sets for Snort were examined carefully and re-implemented to reduce false positives. The monitoring organization was told to take alerts from Snort seriously and take appropriate action in real time.

References:

1. "Backgate Kit analysis and defense" May 2001, Matt Scarborough http://www.incidents.org/react/unicode.php

2. "Norton Ghost information" http://www.symantec.com/ghost/

3. "CERT® Advisory CA-2001-11 sadmind/IIS Worm" May, 2001 http://www.cert.org/advisories/CA-2001-11.html

4. "Vulnerability Note VU#111677: Microsoft IIS 4.0 / 5.0 vulnerable to directory traversal via extended unicode in url (MS00-078)" October, 2000 http://www.kb.cert.org/vuls/id/111677

5. "Microsoft IIS and PWS Extended Unicode Directory Traversal Vulnerability" October, 2000 http://www.securityfocus.com/bid/1806