INCIDENT RESPONSE METHODOLOGY

IRM #3 UNIX/LINUX INTRUSION DETECTION

Live Analysis on a suspected system

IRM Author: CERT SG
Contributor: CERT aDvens
IRM version: 2.0
E-Mail: cert.sg@socgen.com
Web: https://cert.societegenerale.com
Twitter: @CertSG

C'EST VOUS L'AVENIR

SOCIETE GENERALE

TLP:CLEAR

ABSTRACT

This Incident Response Methodology is a cheat sheet dedicated to handlers investigating on a precise security issue.

WHO SHOULD USE IRM SHEETS?

- Administrators
- Security Operation Center
- CISOs and deputies
- CERTs (Computer Emergency Response Team)

Remember: If you face an incident, follow IRM, take notes. Keep calm and contact your business line's Incident Response team or CERT immediately if needed.

→ IRM CERT SG: https://github.com/certsocietegenerale/IRM

→ IRM CERT aDvens (French version): https://github.com/cert-advens/IRM

INCIDENT HANDLING STEPS

6 STEPS ARE DEFINED TO HANDLE SECURITY INCIDENTS

1. Preparation: get ready to handle the incident
2. Identification: detect the incident
3. Containment: limit the impact of the incident
4. Remediation: remove the threat
5. Recovery: recover to a normal stage
6. Lessons learned: draw up and improve the process

IRM provides detailed information for each step of the incident response process. The steps come from NIST Computer Security Incident Handling Guide.

PREPARATION

OBJECTIVE: ESTABLISH CONTACTS, DEFINE PROCEDURES, GATHER INFORMATION TO SAVE TIME DURING AN INCIDENT.

Deploy an EDR solution on endpoints and servers

- This tool became one of the cornerstones of the incident response in case of ransomware or in large scale compromise, facilitating identification, containment, and remediation phases.
- Launch EDR Search and AV scan with IOC explicit rules and get first indicators for remediation progress following.
- Set your EDR policies in prevent mode.

In absence of EDR, a physical access to the suspicious system should be given to the forensic investigator. Physical access is preferred to remote access, since the hacker could detect the investigations done on the system (by using a network sniffer for example).

A **physical access to the suspicious system** should be offered to the forensic investigator.

A **physical copy of the hard-disk** might be necessary for forensic and evidence purposes. If needed, a physical access could be necessary to disconnect the suspected machine from any network.

A good **knowledge of the usual network activity of the machine/server is needed**. You should have a file on a secure place describing the usual port activity, to compare efficiently to the current state.

A good **knowledge of the usual services** is needed. Don't hesitate to ask a Unix/Linux Expert for his assistance, when applicable.

- Use Auditd and Linux Logs like system, message, and applications logs (Apache, NGINX, …)
- Use AppArmor for example

You should have **a regularly updated list of all critical files**, (especially SUID and GUID files) stored in a secure place out of the network or even on paper. With this list, you can easily separate usual SUID files and detect unusual ones.

Have a **map of your usual port activity/traffic rules**.

IDENTIFICATION

OBJECTIVE: DETECT THE INCIDENT, DETERMINE ITS SCOPE, AND INVOLVE THE APPROPRIATE PARTIES.

Unusual Accounts

- Look for any suspicious entry in /etc/passwd, especially with UID 0. Also check /etc/group and /etc/shadow.

- Look for orphaned files, which could have been left by a deleted account used in the attack:

```
# find / \( --nouser -o --nogroup \) --print
```

Unusual Files

- Look for all SUID and GUID files:

```
# find / -uid 0 \( --perm -4000 -o --perm 2000 \) --print
```

- Look for weird file names, starting with ". " or ".. " or " " :

```
# find / --name " *" –print
# find / --name ". *" –print
# find / --name ".. *" --print
```

- Look for large files (here: larger than 10MB)

```
# find / -size +10MB --print
```

- Look for processes running from or to files which have been unlinked:

```
# lsof +L1
```

- Look for unusual files in /proc and /tmp. This last directory is a place of choice for hackers to store data or malicious binaries.

IDENTIFICATION

Unusual Services

Run chkconfig (if installed) to check for all enabled services:

```
# chkconfig --list
```

Look at the running processes (remember: a rootkit might change your results for everything in this paper, especially here!).

```
# ps –aux
```

Use lsof -p [pid] on unknown processes

You should know your usual running processes and be able to figure out which processes could have been added by a hacker. Pay special attention to the processes running under UID 0.

Unusual Network Activity

- Try to detect sniffers on the network using several ways:
  - Look at your kernel log files for interfaces entering promiscuous mode such as :
    “kernel: device eth0 entered promiscuous mode”
  - Use # ip link to detect the “PROMISC” flag.
- Look for unusual port activity:

```
# netstat -nap and
# lsof –i
```

- Look for unusual MAC entries in your LAN:

```
# arp -a
```

- Look for unexpected or new IP addresses on the network:

```
# netstat -ntaupe
# netstat -ant
# watch ss -tt
```

IDENTIFICATION

Unusual Automated Tasks

- Look for unusual jobs scheduled by users mentioned in /etc/cron.allow. Pay a special attention to the cron jobs scheduled by UID 0 accounts (root):

```
# crontab -u root -l
```

- Look for unusual system-wide cron jobs:

```
# cat /etc/crontab
# ls –la /etc/cron.*
```

Unusual Log Entries

Look through the log files on the system for suspicious events, including the following:

- Huge number of authentication/login failures from local or remote access tools (sshd,ftpd,etc.)
- Remote Procedure Call (RPC) programs with a log entry that includes a large number of strange characters …)
- A huge number of Apache logs mentioning “error”
- Reboots (Hardware reboot)
- Restart of applications (Software reboot)

Almost all log files are located under /var/log directory in most Linux distributions. Here are the main ones (paths may vary according to distributions):

- /var/log/message: General message and system related stuff
- /var/log/auth.log: Authentication logs
- /var/log/kern.log: Kernel logs
- /var/log/cron.log: Crond logs (cron job)
- /var/log/maillog: Mail server logs
- /var/log/httpd/: Apache access and error logs directory
- /var/log/boot.log: System boot log
- /var/log/mysqld.log: MySQL database server log file
- /var/log/secure: Authentication log
- /var/log/utmp or /var/log/wtmp: Login records file
- /var/log/syslog: cron, samba activity and more
- /root/.*history: Root user command history
- /home/*/.*history: Users’ command history

To look through the log files, tools like cat and grep may be useful:

```
# cat /var/log/httpd/access.log | grep "GET /signup.jsp"
```

IDENTIFICATION

Unusual Kernel log Entries

- Look through the kernel log files on the system for suspicious events:

```
# dmesg
```

List all important kernel and system information:

```
# lsmod
# lspci
```

- Look for known rootkit (use rkhunter and such tools)

File hashes

Verify all MD5 hashes of your binaries in /bin, /sbin, /usr/bin, /usr/sbin or any other related binary storing place. (use AIDE or such tool)

WARNING: this operation will probably change all file timestamps. This should only be done after all other investigations are done and you feel like you can alter these data.

- On systems with RPM installed, use:

```
# rpm –Va | sort
```

- On some Linux, a script named check-packages can be used.

- On Solaris:

```
# pkg_chk –vn
```

- On Debian:

```
# debsums –ac
```

- On Openbsd (not really this but a way):

```
# pkg_delete -vnx
```

CONTAINMENT

OBJECTIVE: MITIGATE THE ATTACK'S EFFECTS ON THE TARGETED ENVIRONMENT.

- Securely backup important data from the compromised machine, if possible, using a bit-by-bit physical copy of the whole hard disk on an external support. Also make a copy of the memory (RAM) of the system, which will be investigated if necessary.
- Isolate with the EDR and inspect other computers and networks.

Or

- Isolate with the firewall or switches.

If the machine is not considered critical for the company and can be disconnected, shut the machine down the hard way, removing its power plug. If it is a laptop with a battery on, just push the "off" button for some seconds until the computer switches off.

Offline investigations should be started right away if the identification step didn't give any result, but the system is still suspected of being compromised.

Try to find evidences of every action of the attacker: (using forensic tools like Sleuth Kit/Autopsy for example)

- Find all files used by the attacker, including deleted files and see what has been done with them or at least their functionality to evaluate the threat.
- Check all files accessed recently.
- Check log files.
- More generally, try to find how the attacker got into the system. All leads should be considered. If no computer proof of the intrusion is found, never forget it could come from an insider.
- Apply fixes when applicable, to prevent the same kind of intrusion, in case the attacker used a known fixed vulnerability.

REMEDIATION

OBJECTIVE: TAKE ACTIONS TO REMOVE THE THREAT AND AVOID FUTURE INCIDENTS.

WARNING: ONLY START REMEDIATING ONCE YOU ARE 100% SURE THAT YOU HAVE WELL SCOPED UP AND CONTAINED THE PERIMETER - SO AS TO PREVENT THE ATTACKER FROM LAUNCHING RETALIATION ACTIONS.

Temporarily remove all accesses for the involved accounts in the incident and remove malicious files.

RECOVERY

OBJECTIVE: RESTORE THE SYSTEM TO NORMAL OPERATIONS.

No matter how far the attacker has gone into the system and the knowledge you might have about the compromise, as long as the system has been compromised, the best practice is **to reinstall the system completely and apply all security fixes.**

In case this solution can't be applied, you should:

- Change all the system's accounts passwords and make your users do so in a secure way
- Check the integrity of the whole data stored on the system, using file hashes (i.e. SHA256)
- Restore all binaries which could have been changed (Example: /bin/su)
- Replace all compromised packages with safe ones

For more details on authentication and infrastructure recovery, check the Large-scale malware compromise IRMXXX

LESSONS LEARNED

OBJECTIVE: DOCUMENT THE INCIDENT'S DETAILS, DISCUSS LESSONS LEARNED, AND ADJUST PLANS AND DEFENSES.

Report

A crisis report should be written and made available to all of the actors of the crisis management cell. The following themes should be described:

- Initial detection
- Actions and timelines
- What went right
- What went wrong
- Incident cost
- Indicators of compromise

Capitalize

Actions to improve the Unix/Linux intrusion detection management processes should be defined to capitalize on this experience.

Lessons learned

Actions to improve the Unix/Linux intrusion detection management processes should be defined to capitalize on this experience.