Network Device Forensics

Imagine that a rogue laptop connects to your corporate WiFi and is able to access the Internet via your corporate proxy server. Let us assume that your WiFi is protected by a pre-shared key, but that this rogue laptop is owned by a former employee. Will you detect this? And will you be able to trace back to the former employee?

A foreign competitor hires a tech savvy criminal to install a trojaned operating system on your edge router. This trojan facilitates access to your corporate network for unauthorized persons by tampering with the authentication control logic. Will you detect the trojaned router?

These two examples represent two common classes of forensic investigations where forensic evidence needs to be collected from network devices. In the first example, network devices contain evidence of the network traffic that flowed through them. In the second example, network devices have been compromised and forensic evidence needs to be lifted from them.

Forensic evidence gathered by network devices

To operate properly, network devices need to maintain information about the network traffic they process. Since network devices have limited amounts of memory compared to general purpose computers, they tend to collect only the bare essential information for their processes and this information is discarded rather quickly when it is no longer needed.

There is often a significant delay between the time a security incident occurs and the time the forensic investigation starts. And as a switch or a router discards obsolete meta data quickly, you will not find forensic evidence if you react too late.

But you can improve the success rate of your forensic evidence gathering by configuring your switches and routers to collect additional data and persist this data. All professional network devices allow for the logging of events. But the internal event log of network devices is rather small because of the memory constrains. Old events get discarded at a fast rate to make place for new events.

Improve logging, centralized logging

Here is an important first opportunity for you to improve the evidence collection phase of your forensic investigations. Install one or more machines as a central log repository and configure all your network devices to forward events to this central log repository. Dimension your central log repository so that it can hold several months worth of events. The syslog standard is often used to centralize events.

The second opportunity you have to improve the evidence collection phase of your forensic investigations, is by increasing the types of events that are logged, for example DHCP events. Professional network devices classify events by types and by alert level. Not all event types are logged by default, and only events with important alert levels are logged. Increase the type of events, and lower the alert level for event logging. Watch out, you will need to strike a balance between resource usage and log level, because increasing the number of log events has an impact on CPU usage and can thus negatively impact the performance of your network devices.

Utilize on board security features

Make also sure to research security features available in your network devices that can help you indirectly with your forensic investigations. Enable them if necessary. For example, Cisco switches have a DHCP snooping feature. Enabling this feature instructs the switch to build and maintain a table of all successful DHCP transactions it sees passing through its interfaces. This table lists IP addresses, corresponding MAC addresses and the interfaces serving these clients.

Imagine a contractor connects his laptop to your wired network without authorization. You would notice this by monitoring your DHCP logs for rogue machine names. But this will only give you a machine name and a MAC address. This is often not enough to trace back to the contractor. But with the DHCP snooping binding table, you will be able to correlate the IP address and MAC address with a switch interface. This will allow you to find the physical location of the Ethernet connector used by the contractor. Reviewing physical security evidence like access control logs or CCTV images should be enough to identify the contractor. Or you could just ask your employees working near the network access point who used this connection.

In the case of the former employee using your corporate WiFi infrastructure, you would notice this too by monitoring your DHCP logs for rogue machine names. Additional logs from WiFi access points and wireless LAN controllers should enable you to pinpoint the access point used by the former employee. But since WiFi access points do not need a physical connection, you will find it harder to identify the culprit.

Forensic artifacts found in network devices

Network devices can become compromised because their configuration gets modified or because their operating system gets trojanized. Finding forensic evidence for these incidents can become much harder. A secure, centralized log repository is vital so that perpetrators can not erase logs to cover their tracks.

To detect unauthorized configuration modifications, a release management and version control process is necessary. The release management process will make sure that only approved modifications are applied to your network devices, and the version control process will make sure that these modifications are documented. Periodic review of your network device configurations will allow you to detect unauthorized configuration modifications by comparing them with the configurations kept in the version control system. This review process can be automated.

If your network devices support scripting and you have custom scripts, like Cisco IOS Tcl, make sure to include these too in your release management and version control process.

Trojanized devices

The operating system of your network devices can be trojanized in two ways: by trojanizing the operating system files (like Cisco's IOS image files) and booting from them, or by exploiting a vulnerability in the operating system and trojanizing it in memory.

A release management process for network device image files allows you to know if a network device is running an authorized operating system or not. But an unauthorized operating system image is not necessarily a trojanized operating system image. Your success in identifying trojanized operating system images will depend on your network device vendor. For example, Cisco provides lists with cryptographic hashes of all images they release. If the cryptographic hash of the unauthorized operating system image matches a hash in this list, you can be sure that it is a legitimate operating system image and that it is not trojanized. Some high-end network devices can operate with digitally signed operating system images. Periodic review of the digital signature of these operating system images will detect trojanized operating system images.

RAM trojans

But the hardest forensic case to crack is an operating system trojanized in memory. Many professional network devices operate like this: the operating system is stored in a file which is stored on non-volatile, solid-state memory, like flash memory. When the network device is powered on, a very small program stored in ROM will load the operating system from flash into RAM, where it is executed by the CPU. With an operating system trojanized in memory, the image file in flash is intact, but the modifications are made in RAM, where the image file is loaded to be executed. One way to make these modifications in RAM is by targeting the network device with an exploit for a vulnerability[i]. This exploit contains code to modify the operating system in RAM and trojanize it, for example by adding a backdoor functionality.

To investigate such compromise, you need to be able to access and analyze RAM. Cisco IOS has features to access RAM: their routers and switches have a command that allows you to write the content of RAM to a core dump file. This solves the “access” phase of your forensic investigation, but not the “analysis” phase. The structure of the file containing the core dump is not documented. Only Cisco knows the complete details and you will need their cooperation if you need a full analysis. The Cisco Technical Assistance Center (TAC) will sometimes ask clients to provide them with a core dump to help with the analysis of their support cases. But since the RAM core contains everything that was in RAM, it contains a lot of forensic evidence.

But you are not completely dependent on Cisco's TAC for core dump analysis. There are two open source tools that can partially analyze core dumps. The first tool is Cisco Incident Response (CIR) from Recurity Labs GmbH[ii], an open source tool that attempts to detect trojanized core dumps by detecting memory and process anomalies. CIR has been successful in detecting proof-of-concept trojanized IOS images presented at the Black Hat Security conference[iii].

The second tool is the Network Appliance Forensic Toolkit (NAFT)[iv] released by me. It is able to analyze the basic structure of memory and processes, but it is not yet able to automatically detect memory and process anomalies. NAFT is a set of Python programs, it can run on many operating systems. You instruct your IOS device to produce a core dump and transfer it to a tftp server, and then you can analyze this dump with NAFT. For example, command *naft-icd.py processes r870-core* will dump all processes it finds in core dump r870-core:

```
1 Cwe 80049B5C        0        3       0 5552/6000   0 Chunk Manager
2 Csp 80371B90        8      341      23 2640/3000   0 Load Meter
3 Mwe 8118AB24        4     1725       2 5300/6000   0 Spanning Tree
4 Lst 80046D90    14780      841   17574 5484/6000   0 Check heaps
5 Cwe 8004F930        0        1       0 5672/6000   0 Pool Manager
6 Mst 808278AC        0        2       0 5596/6000   0 Timers
```

Pay attention to the fact that although operating systems trojanized in RAM are not persistent (i.e., that rebooting the network devices removes the trojan), network devices are not often rebooted and that the trojan can easily be present for months if not years. And if a trojan runs in RAM with full system access, there is nothing to prevent it from modifying the image in flash to achieve persistence.

Conclusion

There are several preventive steps that you can take to facilitate a forensic investigation of network devices. You can improve the logging of your devices and enable extra information gathering features on your devices. This will help you gather more forensic evidence. Network devices can also become compromised. You can find forensic artifacts in flash and in RAM. There are tools to help you analyze these artifacts.

I hope this article will inspire you to take measures that will facilitate forensic investigations of network devices.

Didier Stevens (Microsoft MVP Consumer Security, SANS ISC Handler, Wireshark Certified Network Analyst, CISSP, GSSP-C, GCIA, GREM, MCSD .NET, MCSE/Security, MCITP Windows Server 2008, RHCT, CCNP Security, OSWP) is an IT Security Consultant currently working at a large Belgian financial corporation. Didier started his own company in 2012 to provide IT security training services (http://DidierStevensLabs.com). You can find his open source security tools on his IT security related blog at http://blog.DidierStevens.com.

[i] Burning the bridge: Cisco IOS exploits, Felix Lindner, http://www.phrack.com/issues.html?issue=60&id=7

[ii] http://cir.recurity.com/

[iii] http://blog.recurity-labs.com/archives/2008/05/27/on_ios_rootkits/index.html

[iv] http://blog.didierstevens.com/programs/network-appliance-forensic-toolkit/