SOLUTION BRIEF

Secure IoT Devices to Protect Against Attacks

The successful distributed denial of service (DDoS) attack sustained against the managed DNS infrastructure at Dyn in October 2016 was the subject of a deep analysis in the *McAfee Labs Threats Report: April 2017*.

The attack was performed using the domain name server (DNS) protocol, which makes it extremely difficult for security technology to distinguish legitimate traffic from hostile traffic. Compounding the problem, attack and legitimate traffic came from millions of IP addresses around the world.

This type of DDoS attack is on the rise, fueled by poorly secured Internet of Things (IoT) infrastructure. The Mirai malware used during the Dyn attack exploited a broad variety of poorly secured IoT devices such as video recorders, printers, surveillance cameras, refrigerators, thermostats, and others. Once an IoT device was infected, the malware spread the infection to other IoT devices, forming a "botnet" and then using their aggregate processing power to execute the DDoS attack.

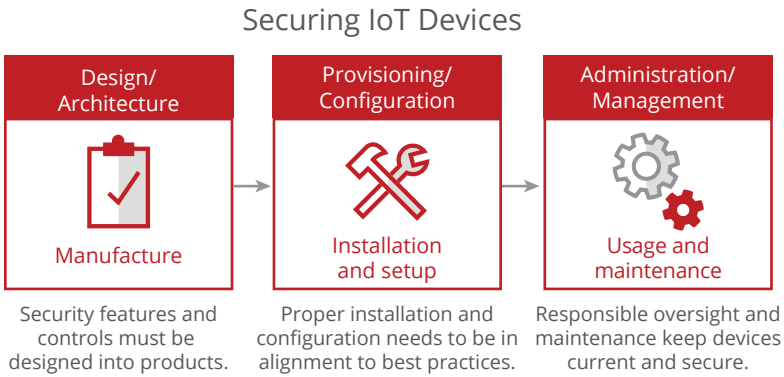
According to the Dyn security team, tens of millions of malicious IoT devices were part of the Mirai-based botnet during the attack's peak.

There is no easy way to conclude whether a network device has been infected or or to determine the stage of infection, which could range from initial steps of code detonation, lateral movement, or control server communication to botnet recruitment for DDoS-orchestrated attacks. There are, however, security recommendations you can follow that can help secure your IoT devices and protect your trusted network.

How to Secure IoT Devices

Attackers follow the path of least resistance to gain control of IoT devices. Usually, this is through weak credentials. But they can adapt to strong credentials and other security controls. This is the pattern we have seen with many attack vectors.

McAfee recommends blocking known exploits and likely future maneuvers by attackers. Take the following three steps to protect IoT devices, from production to retirement:



1. Design IoT devices with security in mind.

IoT manufacturers must embed security into the architecture, interfaces, and designs of their products. Establish and test basic security concepts and capabilities such as compartmentalization of data and code, communication between trusted parties, data protection both in use and at rest, and authentication of users. Products in the future will be more powerful, store more data, and possess more functionality. This means products should have the ability for security updates, feature locking, build validation, software vetting, and default configurations that follow industry best practices. It all starts with the manufacturer; future proofing begins at the foundations. The hardware, firmware, operating systems, and software must be designed to go into a hostile environment and survive. IoT device buyers should examine a potential purchase with this in mind. Has the manufacturer designed and architected the IoT device with security in mind?

2. Securely provision and configure.

Most IoT devices require some kind of setup and provisioning upon installation. Device identity and authentication are an essential part of this two-step process. Proper default configurations that adhere to best security practices are important and should be easy for users to understand. Rules should not allow default passwords, require patches and updates to be signed, data to be encrypted, and only secure web connections. For enterprises, limiting network access, patching in a timely manner, and allowing only approved software to run will go a long way toward keeping IoT devices secure. For gadgets that are capable, implementing security software such as anti-malware, intrusion prevention systems, and even local firewalls will improve a device's defense posture. Detection and telemetry should also be configured to detect when systems are under attack or are functioning in ways not intended by the organization. Policies must be established for privacy, data retention, remote access, key security, and revocation procedures.

3. Apply proper administration and management.

For devices owned by consumers, they alone must maintain the final say in how the device is managed. Manufacturers and online service providers play a role in provisioning, but the owners must retain control of what the devices will do. Provisioning is different than administration. For example, during installation of home cameras, it makes sense to connect to the manufacturer for the latest patches and maybe even set up cloud storage. But customers do not want home cameras controlled by the manufacturers. They should not have the ability to operate devices outside of the buyers' authority. Owners must retain the power to turn on or off their products and choose which online services they allow to connect. This ability requires proper user identification and authentication. Allowing a common default password is not good practice because anyone can take over as the administrator. Imagine if Microsoft Windows came with a default login password for every system. It would create a security nightmare because many users would never change it and attackers could log in as users.

IoT systems must first be able to authenticate their owners. Management functionality must also extend to empower the owners to set limits, data policies, and privacy parameters that are more restrictive than those of any potential third-party vendor. Signed security updates should be automatically installed as they become available. Savvy owners should be able to configure limits for inbound and outbound connections, data types, ports, and security settings. Logs that can be pushed to a trusted system or viewed locally should capture errors, as well as unexpected and unusual activities. A system for remote-warning notifications, via email or text, is a welcome feature on some devices. Finally, a reset capability is required in the event of an unrecoverable compromise or transfer of ownership.

Actionable Policies and Procedures for Securing IoT Devices

- **Research the IoT device's security track record.** Before buying an IoT device, see if it, or the company offering it, has had problems. A quick internet search might suffice. A search of the Federal Trade Commission's website will reveal prior enforcement actions. By performing basic research, you may find that some companies ignore security concerns, while others are more proactive.
- **Keep all IoT device software up to date.** This simple best practice can often remove vulnerabilities, especially those recently discovered and publicly highlighted. Have a patching procedure in place and verify if the patches have been applied successfully.
- **For IoT devices that cannot be patched, mitigate the risk.** You can accomplish this by leveraging application whitelisting, which locks down systems and prevents unapproved program execution.
- **Segment IoT devices from other parts of the network.** Use a firewall or intrusion prevention system. Disable unnecessary services or ports on these systems to reduce exposure to possible entry points of infection. Mirai exploits unused ports.
- **Change defaults and use strong passwords.** Default and weak passwords are the main threat to IoT devices. Adopt good password habits, such as using long phrases, special characters, mixed cases, and digits. Passwords must be strong and not easy to guess.
- **Take advantage of IoT security settings.** Some IoT devices will offer advanced configurations, and you should make the most of them. Certain IoT products may offer separated networking, similar to a guest Wi-Fi network alongside your main connection. That is just one feature—more may come with other products.
- **Connect IoT devices using secure Wi-Fi.** Create strong passwords and use the latest security protocols, such as WPA2.
- **Restrict physical access to IoT devices.** Direct device tampering can also lead to IoT device hacks.
- **Disable Universal Plug and Play (UPnP) support.** Many IoT devices support UPnP, which makes the device discoverable on the internet and vulnerable to malware infections. Disable this when feasible.
- **Power-cycle IoT devices periodically.** Malware is commonly stored in a volatile memory and can be erased by shutting off and restarting the device.

How McAfee Protects Systems and Networks from IoT Device Attacks

In addition to the foregoing list of actionable best practices on IoT devices, McAfee products can help mitigate the risks of malware infections within IoT devices and block the malicious activities of botnets. The following McAfee product configurations can help secure IoT devices and protect systems and networks from attacks coming from IoT devices:

McAfee VirusScan® Enterprise 8.8 or McAfee Endpoint Security 10

- Keep .DAT files up to date.
- Ensure that **McAfee Global Threat Intelligence** (McAfee GTI) is in use; it recognizes more than 600 million unique malware signatures.
- Develop access protection rules to stop installation and payloads of malware:
  - Refer to Access Protection Rules Knowledge Base Article: **KB81095** and **KB54812**.
  - Refer to configuration best practices for McAfee VirusScan 8.8 Enterprise: **PD22940**.
  - Refer to configuration best practices for McAfee Endpoint Security: **KB86704**.

McAfee Host Intrusion Prevention

- McAfee Host Intrusion Prevention can help prevent the spread of malware. Using custom IPS signatures, you can create rules to prevent malware-generated file operations (create, write, execute, read, etc).
- Enable McAfee Host Intrusion Prevention Signature 3894: Access Protection—Prevent svchost.exe executing non-Windows executables.
- Enable McAfee Host Intrusion Prevention Signatures 6010 and 6011 to immediately block the injection.
- Two subrule types accomplish this:
  - 1) Create a custom intrusion prevention system (IPS) signature using the Files engine and a subrule with the following criteria:
    - Name: <insert name>
    - Rule type: Files
    - Operations: Create, Execute, Read, Write
    - Parameters: Include - Files - <path/filename of malware>
      - The filename must include a path. If you wish to wild card the path, begin the filename with "**\" or "?:\" if you wish to wild card the drive letter (for example: "**\filename.exe" or "?:\filename.exe").
      - You cannot use MD5 hashes with the Files parameter, only path/filename.
      - You can also use the drive type to limit the path to a specific drive (for example, hard drive, CD, USB, network, floppy).
    - Executables: Can be left blank, unless you want to limit the signature to specific processes that perform the file operation (for example, explorer.exe, cmd.exe, etc.).
  - 2) Create a custom IPS signature using the Program engine and a subrule with the following criteria:
    - Name: <insert name>
    - Rule type: Program
    - Operations: Run target executable
    - Parameters: <leave blank>
    - Executables: Can be left blank, unless you wish to limit the signature to a specific process as the source executable (for example, if you want to block explorer.exe from running a Target Executable such as notepad.exe).

- Target Executables: Define the executable properties for which you want to prevent execution (for example, if you want to block notepad.exe from running, specify the path/filename of the executable). The executable can be defined using one or more of the criteria (file description, filename, fingerprint, signer).

McAfee SiteAdvisor® Enterprise or McAfee Web Protection

- Use website reputations to prevent or warn users of sites that distribute malware.

McAfee Threat Intelligence Exchange and McAfee Advanced Threat Defense

- McAfee Threat Intelligence Exchange policy configuration:
  - Start with observation mode: As endpoints are discovered with suspected processes, use system tags to apply McAfee Threat Intelligence Exchange enforcement policies.
  - Clean at: Known malicious.
  - Block at: Most likely malicious (blocking at unknown would provide better protection but may also add initial administrative workload).
  - Submit files to McAfee Advanced Threat Defense at unknown and below.
  - Threat Intelligence Exchange Server policy: Accept McAfee Advanced Threat Defense reputations for files not yet seen by McAfee Threat Intelligence Exchange.
- McAfee Threat Intelligence Exchange manual intervention:
  - File reputation enforcement (subject to operation mode). Most likely malicious: Clean/delete.
  - Might be malicious: Block.
- Enterprise (organizational) reputation can override McAfee GTI:
  - You can choose to block an undesired process, for example, an unsupported or vulnerable application.
  - Mark file as might be malicious.
- Or choose to allow an undesired process for "testing":
  - Mark file as might be trusted.

McAfee Advanced Threat Defense

- Detection capabilities:
  - Signature-based detection: McAfee GTI contains more than 600 million signatures.
  - Reputation-based detection: McAfee GTI.
  - Real-time static analysis and emulation: used for signatureless detection.
  - Custom YARA rules.
  - Full static-code analysis: Reverse engineers file code to assess attributes and function sets and fully analyze source code without execution.
  - Dynamic sandbox analysis.
- Create analyzer profiles in which malware is likely to run:
  - Common operatings systems,, Windows 7, 8, 10.
  - Install Windows applications (Word, Excel) and enable macros.

- Provide an analyzer to profile Internet access:
  - Many samples run a script from a Microsoft document that makes an outbound connection and activates the malware. Providing an analyzer profiles an internet connection and increases detection rates.

McAfee Network Security Platform

- McAfee Network Security Platform has signatures in its default policies to detect the TOR network, which may be used to transfer files associated with malware.
- Integrate with McAfee Advanced Threat Defense for new variants of attacks:
  - Configure McAfee Advanced Threat Defense integration in "advanced malware policy."
  - Configure McAfee Network Security Platform to send .exe, Microsoft Office, Java archive, and PDF files to McAfee Advanced Threat Protection for inspection.
  - Verify McAfee Advanced Threat Protection configuration is applied at the sensor level.
- Update callback detection rules (to combat botnets).

McAfee Web Gateway

- Enable Web Gateway anti-malware inspection.
- Enable McAfee GTI for URL and file reputation.
- Integrate with McAfee Advanced Threat Defense for sandboxing and zero-day detection.

VirusTotal Convicter: Automated Intervention

- Convicter is a Python script triggered by McAfee® ePolicy Orchestrator® (McAfee ePO™) software automated response system to cross-reference with VirusTotal any file generating a McAfee Threat Intelligence Exchange threat event.
- It is possible to alter the script to reference other threat intelligence exchanges, such as GetSusp.
- If the threshold for trusting the community is met, the script will automatically set the enterprise reputation. Suggested conviction threshold: 30% of vendors and two majors must agree.
- Filter: "Target file name does not contain: McAfeeTestSample.exe."
- This is a free, community-supported tool (not supported by McAfee).

McAfee Endpoint Threat Defense and Response

- McAfee Endpoint Threat Defense and Response finds and responds to advanced threats. When used in association with threat feeds from McAfee GTI, Dell SecureWorks, or ThreatConnect, new threats can be searched and eliminated before they have a chance to spread.
- Custom collectors allow you to build specific tools to find and identify indicators of compromise associated with malware.
- Triggers and reactions are built by the user to define actions when specific conditions are met. For example, when hashes or filenames are found, a "delete" action can automatically run.

For Further Reading

White Paper: ***More Confidence, Safety, and Security in the Digital World***

Best practices for how to use McAfee Host Intrusion Prevention rules for a malware outbreak: **KB84507**

SIEM orchestration. How McAfee Enterprise Security Manager can drive action, automate remediation, and increase situational awareness: **PD24830**

White paper: ***Secure Beyond the Signature***

FAQs for McAfee Network Security Platform. advanced malware detection: **KB75269**

McAfee Web Gateway Product Guide. Web filtering: **PD26339**

McAfee
Together is power.

2821 Mission College Boulevard
Santa Clara, CA 95054
888 847 8766
www.mcafee.com