Linux Servers: Why Native Security is Not Enough

This document explores the realities of Linux vulnerabilities and outlines how additional security, like Trend Micro Deep Security, can help to secure Linux servers.
OVERVIEW

In threat discussions, we often hear these common questions, both online and with prospective customers:

"There are no threats for Linux servers. Aren’t they built to be secure?"

"Linux servers are so secure and hardened, why do we need security controls on those?"

"I do understand there are threats out there but I am not aware of any major attacks on Linux servers"

To better understand the threats a Linux server faces, it is important to look at them under the lens of both attack avenues and persistence. Most commonly, an attacker will leverage a network vulnerability to gain access and leave behind a mechanism for later access. However, in the case of direct attacks like ransomware, persistence is not needed so protection beyond the network is needed. To help clarify the need for more than native security, this whitepaper will walk through the threat landscape for Linux servers, and highlights key security controls that can make a Linux server secure.

LINUX IS SECURE…RIGHT?

While the Linux operating system is a robust choice for supporting server deployments, simply deploying it is not a panacea for security. Just like Microsoft Windows, there are vulnerabilities that can be exploited in both the operating system (kernel) as well as the application stack that runs on the server. At the operating system level, major distro vendors regularly publish details on security issues with their platform. Examples include:

- https://access.redhat.com/security/
- https://www.suse.com/support/security/
- https://www.ubuntu.com/usn/

For staying on top of Linux threats, a good resource is http://www.linuxsecurity.com/, where regular updates and disclosures are shared for Linux as a platform.

At the application level, for example, the LAMP stack is affected by dozens of vulnerabilities and the recent Apache Struts 2 vulnerability was being exploited independent of the operating system. Foundational elements such as GNU glibc and OpenSSL are other examples of code which left Linux (and other systems) vulnerable. For reference, a sample list of Linux vulnerabilities can be found in Appendix 1.

It is very important to not confuse vulnerabilities with threats. While there may be fewer known threats for Linux, if you look at the National Vulnerability Database, there are a similar number of vulnerabilities reported for both Linux and Windows operating systems:
PROTECTING AGAINST NETWORK THREATS WITH INTRUSION PREVENTION (IPS)

An intrusion prevention system (IPS) protects against vulnerabilities in core operating system AND the application stack running on top. Great examples of network-accessible vulnerabilities with wide-spread impacts are Heartbleed and Shellshock, but there are many more. And even though Shellshock has been in the wild since 2014, there are still many (over 180,000) publicly accessible servers that have the vulnerability!

If you run a web server on Linux (running at least 37% of the web servers out there according to W3Techs), you need protection against vulnerabilities affecting them, including Apache, Nginx, etc. With many vulnerabilities available and no protection in place, attackers can upload and execute arbitrary code, including installing backdoors, removing/deleting business-critical files, or encrypting the files on the server in a ransomware attack.

Examples of weaponized exploitation of some application-level vulnerabilities include:


With more and more servers moving beyond the enterprise boundary and into the cloud, network protection at the host-level becomes increasingly important, as workloads need to defend themselves vs. having a perimeter around them.

To give a better understanding of how Trend Micro helps, Table 1 shows number of relevant vulnerabilities protected by Deep Security. These vulnerabilities affect the core operating system and core services like bind, OpenSSL, Samba etc. and also vulnerabilities in other various applications that run on these platforms.
DO I NEED ANTI-MALWARE FOR LINUX?

Contrary to popular belief, there is a lot of malware for Linux platform. While the numbers in comparison to Microsoft Windows are not as high, there are still tens of thousands of pieces of malware designed for Linux. Threats like BASHLITE, Mirai, SAMSAM, Umbreon, LuaBot are some examples and notably, the Mirai botnet is a unique case where the Windows version of the malware came out months later and the botnet had established a large footprint already with the Linux variant. For details on some of these please refer to:


In addition to the need to defend against Linux malware, another common use case for "Linux AV" is on file servers hosting Microsoft Windows files. It’s not uncommon for a Linux server to host Windows files and the onus to scan them is on the Linux host.

Aligned with leading industry analyst firms like Gartner¹, Trend Micro agrees that deploying ONLY anti-malware is inadequate for protecting servers. However, most attacks on datacenters that lead to breach involve the installation of malware as part of the attack chain. This is why compliance and security frameworks such as PCI-DSS (Section #3), SANS CIS Critical Security Controls (Section #8), and NIST Cybersecurity Framework (Section DE.CM-4) all continue to recommend anti-malware as a best practice.

OTHER WAYS TO PROTECT LINUX SERVERS

Beyond IPS and anti-malware, there are multiple security controls that can be very effective in both protecting servers/applications, as well as help with the compliance challenge. These include:

- **Application Control**: Most servers have a limited well defined 'task' or role, and technically, this should translate into known or deterministic processes and scripts being run on that host. Anything else should be considered highly suspicious. Application control helps 'lock down' the host to prevent any unknown process or script from running. This prevents the malware from running in the first place or attackers from taking advantage of backdoors that it might have placed on the server.

---

• **Integrity Monitoring:** A typical production server does not go through many changes, except during the change management process window. A new threat is likely to make changes to the system, so it’s important to watch for these. Integrity monitoring helps with monitoring the system for any changes outside of the change window. It’s important to note that it is not always a binary file that gets introduced to a computer; the malware or attack could simply modify the configuration files (e.g. `/etc/hosts` file) or it could simply upload a Web shell for persistence. With integrity monitoring, any malicious activity can be flagged quickly as events from known binaries provided by the OS/application vendors can be whitelisted.

In addition, Linux workloads are often subjected to brute force and password/key stealing attacks. Techniques like integrity monitoring and log inspection help to detect malicious or unauthorized changes.

• **Log Inspection:** Log Inspection simplifies log management for audit and threat detection. It helps bubble up the most important logs first. Scanning log files and having a continuous monitoring process helps identify threats early in the cycle. Attacks like SQL Injection, command injection, attacks against APIs can be seen in the logs and then action taken.

**DEEP SECURITY: SECURING HYBRID CLOUD WORKLOADS**

As organizations deploy Linux (and others) servers to new environments, Trend Micro Deep Security can play a significant role in addressing many of the critical security requirements. Delivered from the market leader in server security\(^\text{2}\), Deep Security streamlines operations through its ability to secure workloads across physical, virtual, cloud, & container environments. Available as software, service (PCI DSS Level 1 certified), or via the AWS and Azure marketplaces, it can help organizations streamline the purchasing and implementation of the essential security elements recommended by SANS. With proven API-level integration with VMware, AWS, and Azure, Deep Security provides full visibility across the hybrid cloud, and includes the ability to automate security aligned with DevOps approaches.

Trend Micro Deep Security, powered by XGen™, is a host-based security control product that secures millions of servers across thousands of customers around the world. It includes a cross-generational blend of security controls for protecting servers, including:

- **Network security** enabling virtual patching, network attack prevention, and lateral movement prevention through Intrusion Detection & Protection (IDS/IPS) and a host-based firewall

- **Anti-malware** with Web reputation and sandbox integration to protect vulnerable systems from the latest in threats.

- **System security** through application control, integrity monitoring & log inspection, enabling the lock-down of systems, discovery of unplanned or malicious changes to registry and key system files, as well as discovering anomalies in critical log files.

---

\(^2\) Worldwide Endpoint Security Market Share, 2015, IDC #US41867116 [Published October 2016]
CONCLUSION

With thousands of customers and millions of servers protected, Trend Micro Deep Security is designed for securing Linux (and other) servers across the hybrid cloud, delivering a cross-generational blend of threat defense techniques in a single product that has been optimized for securing physical, virtual, cloud, and container workloads. It includes protection from advanced attacks like ransomware and multiple capabilities in a single product that allows for vendor consolidation and the simplification of operations without compromising security. Ranked number one in market share by IDC and positioned furthest for completeness of vision and highest for ability to execute in the leadership quadrant of the Gartner Magic Quadrant for Endpoint Protection Platforms, you can feel confident in choosing Deep Security to protect Linux servers across your hybrid cloud deployments.

APPENDIX 1

Linux vulnerability examples:

- “Most serious” Linux privilege-escalation bug ever is under active exploit (Dirty Cow) | Ars Technica
  https://arstechnica.com/security/2016/10/most-serious-linux-privilege-escalation-bug-ever-is-under-active-exploit/

- "The code-execution bug resides in the Apache Struts 2 Web application framework and is trivial to exploit. Although maintainers of the open source project patched the vulnerability on Monday, it remains under attack by hackers who are exploiting it to inject commands of their choice into Struts servers that have yet to install the update, researchers are warning. Making matters worse, at least two working exploits are publicly available."

- DROWN - Cross-protocol attack on TLS using SSLv2 (CVE-2016-0800)
  https://access.redhat.com/security/vulnerabilities/drown

- Linux bug leaves top websites vulnerable to serious hijacking attacks (RFC 5961)
  https://arstechnica.co.uk/security/2016/08/linux-bug-off-path-attack-website-hijack/

- 'Bash' command flaw leaves Linux, OS X and more open to attack

- Linux vulnerability leaves thousands open to DNS attack
  http://www.itpro.co.uk/security/26060/linux-vulnerability-leaves-thousands-open-to-dns-attack