WHITE PAPER

BeyondTrust™

VISIBILITY. KNOWLEDGE. ACTION.

How Secure is Your sudo?

The Good, the Bad, the Ugly of Open Source Privileged Account Management

October 2015

Table of Contents

For the latest updates to this document, please visit:
http://www.beyondtrust.com

Warranty

This document is supplied on an "as is" basis with no warranty and no support.

Limitations of Liability

In no event shall BeyondTrust be liable for errors contained herein or for any direct, indirect, special, incidental or consequential damages (including lost profit or lost data) whether based on warranty, contract, tort, or any other legal theory in connection with the furnishing, performance, or use of this material.

The information contained in this document is subject to change without notice.

No trademark, copyright, or patent licenses are expressly or implicitly granted (herein) with this white paper.

Disclaimer

All brand names and product names used in this document are trademarks, registered trademarks, or trade names of their respective holders. BeyondTrust is not associated with any other vendors or products mentioned in this document.

Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.

A Brief History of Unix and Linux - The Problem with Root

In the 1960s, Unix was created by a multi-organizational effort to develop a dependable time-sharing operating system. The Unix system fostered a distinctive approach to software design – solving a problem by inter-connecting simpler tools, rather than creating large monolithic application programs. The Unix development and evolution led to a new philosophy of computing, and it has been a never-ending source of both challenges and joy to programmers around the world.

After four decades of use, the Unix OS is still regarded as one of the most powerful, versatile, and flexible operating systems in the computer world. Its popularity is due to many factors, including its ability to run a wide variety of machines, from micros to supercomputers. IT also fell in love with its portability, all of which led to Unix's adoption by many manufacturers. Universities began using Unix for research, but over the years enterprises embraced Unix and began using Unix to run databases and other business applications.

As time went on, more and more business critical assets were being managed by Unix. Because universities used Unix to create a collaborative environment, access to high-privilege administrative accounts in Unix was based on little more than trust. While this made sense to education users, it had the opposite effect on the business world where protecting access to confidential information is vital to survival.

Over the last four decades, the foundation of IT systems management has been built on the concept of the administrator. Whether root on a Unix or Linux system, or a DBA or Windows administrator, the administrator role gives the user the power to configure virtually every aspect of a system. Even though the administrator role is considered to be the highest level of privilege, access to such privilege is protected by the simplest of controls, such as knowledge of a root password. The root or administrative account has historically been shared among a group of trusted individuals, making it virtually impossible to track the actions of any specific user of this group. Though this scenario was ideal for university use, the IT risk to enterprise use was obviously problematic.

The Birth of sudo

In 1994, Todd Miller released an open source privileged account management solution called sudo. sudo (superuser do) allows a system administrator to work using his own account and switch to root or other user identity available on the system only for commands that they need to perform. sudo was also designed to improve logging to see what actions were being performed by specific individuals.

Many organizations have attempted to use sudo to authorize users to run commands without knowledge of the root password, and a lot of IT organizations praise sudo to their managers, CISOs and CTOs.

On the flip side, other enterprises have found drawbacks with sudo because sudo was not designed for an enterprise mainly in terms of compliance. Hence, an IT-divide of passionate

opinion regarding sudo has evolved over the last 15 years. Many users focus on sudo's feature set and low purchase cost as supporting factors. However, other company stakeholders such as management, compliance auditors and analysts have also raised red flags regarding sudo that larger or growing companies need to be cognizant of at the very least.

Whether a stakeholder is just discovering open source or is a veteran, one will or has come across some of the gotcha's, quirks, and wow-type experiences. It can be a roller coaster ride of sorts. If one is seasoned enough to navigate and tread the enormous landscape of open source projects, it can be positive depending upon the IT environment one exists in.

Generally, people have been hard-coded to think of technology in one perspective, and open source goes against all the grains in our post-modern capitalistic society. Open source yells, "I want to be free for the good of mankind, and I want to unite developers and enthusiasts."

On the other hand, some old adages still ring true, such as "Sometimes you get what you pay for, and sometimes if it sounds too good to be true, then it usually is." The war of opinions over sudo all have valid points, and that is why it is important to look at the good, the bad and the ugly of from an objective position while you are analyzing.

sudo – The Good

IT IS FREE

This is generally one of the most important factors for an enterprise evaluating sudo as a viable solution for Unix privileged account management. If judging a book by its cover, sudo's initial cost is always appetizing to security budgets.

ADDITIONAL LAYER OF PROTECTION TO ROOT

Too often, organizations fail to achieve necessary milestones in the control of root access to the heart of IT and its critical assets. Instead, they simply assign access to a shared account where activity may be difficult if not impossible to correlate to the actions of a single user. With sudo, the root account password does not need to be shared with everyone who needs to perform tasks on Unix and Linux systems.

ADDITIONAL LAYER OF PROTECTION OVER ACCIDENTS

Before sudo, any mistakes, significant or minimal, were much harder to pinpoint and access to shared accounts was virtually impossible to audit for accountability over what was actually done in the system. Though limited in its ability, sudo does help prevent some really horrible things that happen when administrators work as root and try and complete tasks in a hurry.

SIMPLIFIED PROCESS FOR ADMINISTRATORS

Though this is often argued as a security and compliance risk, sudo can be configured in such a way that members of a designated group do not need additional authentication to become root, which results in higher productivity.

UPGRADE IN AUDITING OVER NATIVE UNIX SYSTEMS

Auditing in Unix and Linux was never a priority under the original intentions of these operating systems, since its origin of use was in academic research programs and databases. Though sudo's logging functionality has serious security and compliance red flags, it is definitely a step in the correct direction to log what actions a user performed with privileged access.

TIME-TICKETING SYSTEM

When a user invokes sudo and enters their password, they are granted a "ticket" for five (5) minutes (default time). This time-ticketing system give some level of protection in situations where there is a danger of leaving a root shell open where others can physically use an administrator's keyboard.

sudo – The Bad

GENERATED DATA IS INSECURE

As security is provided through the binaries and configuration files, sudo lacks the necessary tools required to protect the integrity of generated security data.

IT CONTROL IS AN ILLUSION

sudo has the ability to invoke a shell that has no control over itself whatsoever. This means that an experienced user could obtain full-access over valuable systems and applications. sudo's control file delegates privileged commands based on user, group, host, command, and command arguments. Many organizations write policy files based on these filters, but that does not always make an organization compliant. To ensure proper control, many other filters should be applied to policies.

THERE IS NO OFFICIAL SUPPORT AVAILABLE

Since sudo is open source, there is no directly accountable contact to call upon if an organization encounters technical problems with sudo. However, there is a very large sudo community that collaborates over common problems and solutions, and many people work very hard to help others with issues. Still, the unpredictability of researching and retrieving a correctable solution in a timely manner translates to lower productivity and higher cost in managing privileged identities.

LOG SECURITY AND COMPLIANCE QUESTIONABLE

While sudo logs and tracks user activity over privileged accounts, it is very difficult to ensure safety of sudo's logs. sudo does not natively protect log records from alteration, and cannot provide remote logging to remote servers, which are best practices for security and compliance.

LACKS INDEPENDENT QA TESTERS

When an organization is seeking to protect assets worth significant amounts of money, it would be prudent to ensure that the solution they are implementing has been tested and validated regarding the security and functional features. Though there are many contributors to issues and features, there is no dedicated group of QA testers to provide assistance and assurance that the product is enterprise ready.

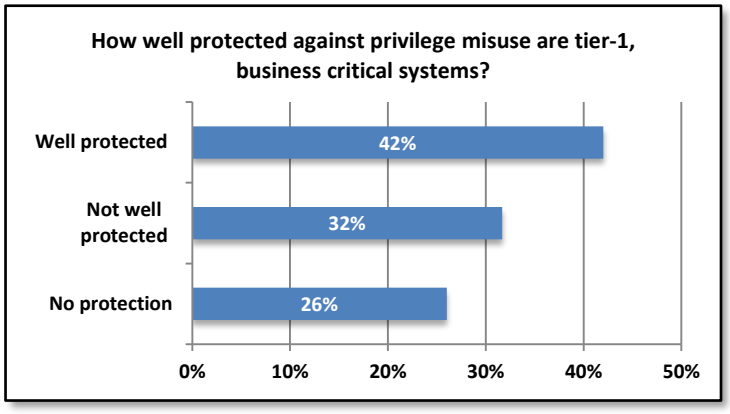
SUDO IS NOT DESIGNED FOR THE ENTERPRISE

Can sudo work in an enterprise? Yes. But, was sudo really designed for the enterprise? Looking at the history of sudo, it is safe to say, "No." Since sudo was not intended for large scale deployments, there simply is not enterprise development discipline needed in security solutions. sudo is a good stepping stone for smaller scale environments, but lacks architectural vision or general security of code that could be used to protect critical assets. When sudo is deployed with the enterprise under consideration where the sudoers file is centralized using an LDAP server, consistency can be had and enforced. When policies are enforced based on user names, well defined security policies can be put in place most of the time. However, when policies mix the security considerations (i.e., using usernames and group names), an organization can end up with conflicting policies and corrected by a potentially ever-growing list of constraints. This can lead to maintenance issues and a weakened security environment.

sudo – The Ugly

CISOS DO NOT KNOW HOW EFFECTIVE THEIR SECURITY CONTROLS ACTUALLY ARE

In the current environment, business critical, tier-1 applications are attractive targets for adversaries. Accessing privileged user credentials for these resources can provide access to ecommerce data, ERP systems managing employee data, customer information, and sensitive financial data. For those systems (for example Linux and Unix servers), 58% of respondents to a survey of 728 IT professionals conducted in January 2015 by BeyondTrust believe their current controls against misuse are inadequate, immature or non-existent. Clearly, addressing this shortcoming should be a priority for these organizations.



Security managers need to develop a process for tracking key performance indicators that measure effectiveness of their data security programs, such as sudo, which includes metrics as frequency and cost of incidents. Meeting these security and compliance mandates using sudo can become very ugly for auditors and administrators and raise cost significantly, even though sudo is initially a cost-saving solution.

REAL-WORLD EXAMPLE OF SUDO'S UGLY SIDE

CETREL S.A. (www.cetrel.lu), a leader in advanced electronic payment technology, expert in electronic transfers, and a trusted partner for electronic payment offers, experienced significant compliance and auditing challenges using sudo to manage their IT environment.

Nicolas Debeffe, head of operational security at CETREL, is responsible for overseeing CETREL's security operations which includes their complex IT environment. For the last several years, Mr. Debeffe's security team had been using sudo to manage their critical Unix and Linux assets and trace any access from CETREL's support teams to applicative or generic users.

While sudo initially seemed to manage CETREL's IT environment, they soon discovered that there was an imminent need to find a simpler and more secure method to manage access and accountability to generic users.

"As we have been continually adding Unix and Linux servers to our environment, as required for our operations, it was clear sudo raised significant red flags over the adequate security over our logs required by PCI DSS mandates," said Nicolas Debeffe.

"Productivity was being hindered, as reviewing sudo logs required accessing every server individually. Furthermore, sudo logs were alterable by the super user and the sudo configuration time required by system engineers was simply unacceptable, added Debeffe."

This example is a very common and real challenge for security managers globally, and the faster organizations are cognizant of such red flags, the faster preventative measures can be implemented from a strategically and compliant perspective.

The True Cost of Free

Key questions that should be asked when evaluating privileged account management solutions – whether open source or commercial – should include:

1. What is the cost of the solution?
2. What is the cost to implement the solution effectively in our IT environment?
3. What is the cost to manage and audit such tools, such as:
    a. man-hours to enforce policy changes across network
    b. man-hours to add new environments

c. man-hours to code reports

d. audit costs related to security and compliance

4. Has the solution been properly tested and vetted to meet security and compliance?

Privileged account management is critical business systems, and if not managed correctly, can introduce significant compliance, security and productivity risks as shown in this white paper. It is always good to ask, "What is the true cost of this solution?"

Bridging the Gap from sudo with BeyondTrust

Making the business case for transitioning away from sudo to a commercially supported solution or changing the way sudo is managed within your organization primarily revolves around three themes:

- **Lack of support for sudo and the need for third-party tools could be more resource intensive (expensive):** Free isn't free in terms of man hours. There is no real support for sudo, it's more resource intensive to manage it, and it requires extensions and other third-party tools for ease of use.
- **Lack of compliance capability and reporting:** sudo has gaps in meeting most compliance requirements; reporting and analytics are rudimentary.
- **Lack of security and scalability:** Policy controls in sudo are lacking, it doesn't scale well and has real security vulnerabilities that are difficult to patch.

Compared to the freeware sudo, BeyondTrust offers better compliance and security capabilities. PowerBroker for Unix & Linux allows IT organizations to efficiently delegate Unix and Linux privileges and authorization without disclosing passwords for root or other privileged accounts. The solution enables IT to record all privileged sessions for audits, including keystroke information. With it, organizations can achieve privileged access control requirements without relying on native tools or sudo. As well, PowerBroker for Unix & Linux is more highly scalable, features integration for reporting, discovery and a dashboard for management, and DVR-style session recording for compliance.

But replacing sudo is not an easy task. Once organizations have made the decision to start using a commercially-supported solution they often find themselves sacrificing their multiple sudoers files for the sake of centralization. Using PowerBroker for Sudo in conjunction with PowerBroker for Unix & Linux can help make the transition easier by enabling organizations to maintain sudo more efficiently. PowerBroker for Sudo may even prove suitable for long term usage on low priority servers or in area where replacing sudo completely is not feasible.

Although PowerBroker manages policies centrally and sudoer files are managed separately, BeyondTrust provides a way to centralize multiple sudoers files. PowerBroker accomplishes this by implementing a centralized repository with change management functionality,

providing storage for sudo/sudoers configuration for all individual hosts with an enhanced policy grouping to more easily define user roles across the enterprise.

This approach provides a central database, sudo policy, management, logging and change control while providing a simplified transition strategy for customers looking to move from sudo to a full commercially-supported solution.

Take the Next Step

A thorough understanding of the good, bad and ugly of sudo is necessary in weighing the benefits of transitioning to a commercially-supported solution. For non-critical systems, maybe sudo is enough. For critical servers, you will likely need full privilege delegation capability. Regardless, BeyondTrust can address both needs with an integrated solution that centralizes policies and logging in a single interface.

Contact BeyondTrust today to get a one-to-one demo or trial the solution for yourself.

About BeyondTrust

BeyondTrust® is a global security company that believes preventing data breaches requires the right visibility to enable control over internal and external risks.

We give you the visibility to confidently reduce risks and the control to take proactive, informed action against data breach threats. And because threats can come from anywhere, we built a platform that unifies the most effective technologies for addressing both internal and external risk: Privileged Account Management and Vulnerability Management. Our solutions grow with your needs, making sure you maintain control no matter where your organization goes.

BeyondTrust's security solutions are trusted by over 4,000 customers worldwide, including over half of the Fortune 100. To learn more about BeyondTrust, please visit www.beyondtrust.com.