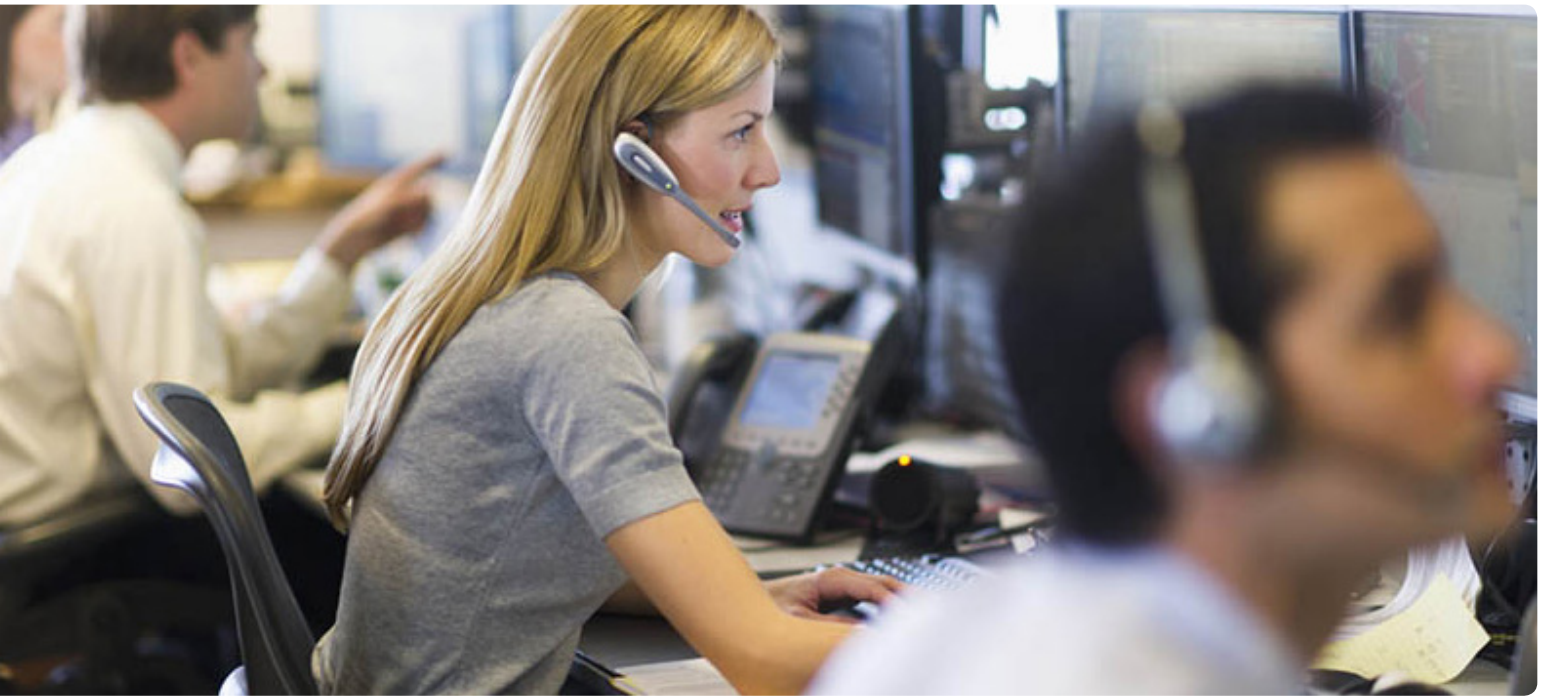
Controlling and Managing Superuser Access

A Primer on Privileged Account Management

Written by Kris Zupan, chief architect, Dell Software



Abstract

Effectively managing privileged accounts (sometimes called superuser accounts) is becoming more and more critical as security and compliance emerge as the driving forces behind most IT initiatives. Unfortunately, native tools and manual practices for privileged account management are proving to be inadequate for today's complex heterogeneous enterprise. This white paper explores the risks associated with privileged accounts, and explains how solutions from Dell® mitigate those risks by enabling granular access control and accountability while preserving necessary access and ease of use.

This paper is intended for CIOs, IT directors and managers, security and compliance officers, and administrators in enterprises of all sizes, especially those who have not established firm control over all of their organizations' privileged user accounts.

The challenges of privileged accounts

Privileged accounts are necessary but risky

Privileged accounts (known as "root access" in the Unix world) are necessary from an administrative perspective. Administrators need easy access to certain areas, and sometimes the only way to conveniently gain that access is to have privileged accounts—that is simply how some operating systems work. While operating systems have become significantly more powerful in recent years, privileged access has not evolved as quickly, so a single, all-powerful level of access still exists in many enterprises. For instance, many Unix administrative tasks can't be carried out without root access, and many of those tasks are quite routine. While a small business may have only a single trusted person with privileged access, most midsize to large businesses have multiple privileged administrators.

A surprisingly large number of people often wield incredible power within the native OS—much of which is unnecessary for each individual to fulfill his or her role.

The problem is that operating systems do not natively offer a way to discriminate more granular privileged access: it's an all-or-nothing proposition. Therefore, a surprisingly large number of people can often wield incredible power within the native OS—much of which is unnecessary for each individual to fulfill his or her role. Since privileged accounts can be used to bypass standard controls and authorization levels, a person with a privileged account often has unlimited access and may be able to inflict significant damage to networks, servers, applications and data.

The problem is not only that too many administrators can inflict damage. Making matters worse is that administrators may be able to work outside the network's identity management system and hide their actions. Most organizations face serious challenges in analysis and discovery of security breaches, both in real time and after the fact. They have problems finding out what went wrong, who did what and when they did it. This opens up a level of risk that has no place in a secure IT environment.

The solution lies in a combination of policy, checks and balances, and automated oversight that can enable more granular access.

Solving your privileged account management challenges

Establishing checks and balances

In the United States' system of government, constitutional checks and balances assign separate powers to the judicial branch, legislative branch and executive branch. Think of the executive branch (Office of the President) as the privileged account holder; the president wields the ultimate access rights and decision-making authority—but that power is mitigated by the oversight of the other two branches. In an enterprise environment, a similar system of checks and balances can be established to limit the power, authority and access rights of privileged users.

In most cases, granting the "keys to the kingdom" to a single person is not really necessary—the operating system's privileged account system does not have to be used as is. A more granular delegation of authority, policy-based control and automated workflows can impose an added layer of security over an inherently insecure designation while still enabling administrators to get their jobs done efficiently and effectively.

An optimal approach to privileged account management should include the following checks and balances:

- Privilege safe
- Command control and granular delegation
- Keystroke logging and session audit

Through a policy-driven implementation of these areas, an organization can protect its data, prevent security breaches and ensure compliance with an ever-widening array of rules. This is the realm of *privileged account management:* a combination of processes, policies and technologies that ensure that privileged users and superusers who share administrative credentials are doing the right things, that access is delegated on an as-needed basis and that an audit trail is kept in place at all times. In short, privileged account management adds accountability back into what is otherwise a free-wheeling and overly broad system of administrative access.

Managing privileged accounts: Who guards the guards?

The fundamental challenge of account management is that the IT department is usually delegated the role of managing access, authentication and authorization—but often no one imposes that control over the IT department. Those who have privileged access may have a common ethos for sharing information (including passwords), self-policing their actions and keeping that enhanced level of access to themselves. But who guards the guards?

It's all too common for enterprises to lack any coherent strategy for privileged access. Most large organizations have multiple administrators, including Windows, Unix and other administrators, each with his or her own tasks to complete. IT administrators have a particular culture of trust between themselves, and it's not unusual for multiple administrators to share a single superuser password. This is a friendly way to do business, but an unnecessarily risky one.

IT people don't always appreciate what management sees as essential: the need to impose strict controls over themselves. As a result, something exceedingly powerful—the unlimited access that can be gained from privileged accounts—receives little oversight and is too often protected by just a cobbled-together, ad hoc, informal and frequently ignored set of administrative protocols. As a result, in any enterprise with more than a handful of IT staff, a number of people will have privileged access, which allows them to do just about anything they wish.

The basic challenges companies face with regard to privileged access include:

- No accountability whatsoever, since there is (amazingly) no oversight or management system to control privileged access
- Too many people with access to superuser or root accounts, including admins with limited responsibilities gaining unnecessary access to functions they don't need
- Lack of control over the privileged access password, which is frequently shared

Safeguarding sensitive information and ensuring compliance

Protecting sensitive data and applications is a growing concern for CIOs and senior IT management. Faced with a rapid growth in traffic and data volume, both small and large organizations have a need for privileged users with broad access to servers. This naturally makes it difficult to protect data and comply with regulatory mandates.

Controlling third-party vendor and consultant access

Technologies such as cloud computing, along with globalization and the economic recession, have transformed the ways that enterprises conduct business. In particular, more organizations are using third-party vendors and consultants to acquire specialized solutions without the burden of adding full-time IT staff.

While this new business model brings many benefits, it also creates challenges, the greatest of which is secure access. Third-party specialists require access to the corporate network, and in many cases, this must include a level of privileged access. Granting privileged access to people inside the company brings enough problems; providing privileged access to outside consultants, who are running computers that your organization has not provisioned and that may not be firewalled or protected from malware, is a disaster in waiting. Even a conservative security policy would prohibit such access, but a strict prohibition would make it too difficult for contractors to get the job done.

The only good solution is to enable remote vendors with a controlled version of privileged access, so that they can gain entry to those areas they truly need to access without being able to snoop around the entire network looking at confidential corporate data assets.

Understanding the risks: When things go wrong

The risks of privileged accounts are not just theoretical. Disgruntled and terminated superusers have been known to steal or sabotage data on their way out the door, and industrial espionage occurs on a regular basis. Identity theft and theft of corporate secrets takes place more frequently than many people are aware, and it's often an insider who is the culprit.

Privileged account management is a combination of processes, policies and technologies that ensure that privileged users and superusers who share administrative credentials are doing the right things, that access is delegated on an as-needed basis and that an audit trail is kept in place at all times.

The unlimited access that can be gained from privileged accounts usually receives little oversight and is too often protected by just a cobbled-together, ad hoc, informal and frequently ignored set of administrative protocols.

Such was the case at a Fannie Mae facility in Urbana, Maryland on October 24, 2008, when a contract administrator was terminated and decided to take revenge by planting an electronic "time bomb" designed to propagate throughout the Fannie Mae network of computers and destroy all data. According to a federal indictment[1], five days after the contractor had been dismissed, an engineer discovered a malicious script embedded in a routine program. The disaster was averted, but the situation could just as easily have gone the other way. The vulnerability was that the agency lacked a procedure for disabling privileged access immediately upon a person's separation. The rogue employee still had access until later the same evening, giving him enough time to sow seeds of destruction.

Requirements for a privileged account management solution

An alternative to giving administrators unlimited access with no oversight

Clearly, administrators need to have access to do their jobs, but the "all or nothing" approach native to the OS is inadequate and outdated. Most admins who have privileged access do indeed need privileged access to one or more areas of the network, but it is unlikely that they need privileged access to everything. A method to allow easy, unfettered access to what is needed, when it is needed, mitigated by restricted access to what is not needed, would solve the problem. Such a system would:

- Delegate specific privileges to administrators based on role.
- Include a policy engine that impartially delegates access based on need.
- Provide a complete audit record with full details of access and specific actions taken.

Traditional approaches to privileged account management are almost always inadequate

Motivating a group of IT people to adhere to a new management policy is a little like trying to herd cats. Upper management often has a hard time trying to impose its point of view over the IT department. It sometimes just can't be done; IT people are an independent-minded fraternity. Managing IT from outside the department is difficult because management doesn't really understand everything that IT people do. When the operations manager, or any manager from outside IT, steps in and announces, "We need to restrict your access," that outside person had better be armed with a very compelling argument and a firm resolve.

Basic conflicts occur when management views privileged or unlimited access as a problem while admins see it as standard operating procedure. As a result, organizations tend to adopt one of three solutions:

- Issue a memo, which everybody understands will be uniformly ignored, but management has been placated.
- Implement a manual solution (often called a "firecall ID") that involves writing the privileged access password on paper, sealing it in an envelope and storing it in a secure, physical location (such as a safe) controlled by an outside trusted employee or manager. That outside individual is tasked with changing the password each time it has been used.
- Create individual solutions and policies that lack unification, solving only one problem at a time.

The first approach above is clearly inadequate. The second solution attempts to address the issue, but because it is primarily human-controlled, it is still subject to error, loss and

[1] Department of Justice, U.S. Attorney's Office for the District of Maryland, January 27, 2009, "Former Fannie Mae Contractor Employee Indicted for Computer Intrusion."

intentional misuse. In addition, this approach breaks down when there are dozens or hundreds of accounts at hand.

The third solution may be adequate in smaller company environments. The open-source solution sudo, for example, solves a lot of problems and may be all that is needed if there are only a handful of Unix and Linux servers involved. But for larger installations, sudo offers no centralized management function to control multiple servers from a single management console, nor does it provide an audit trail. (For more on sudo, see the section "The sudo project" below.)

Three basic policies are essential to success

Preventing disasters like the Fannie Mae incident described above is not rocket science, but most companies simply don't do it. That rogue administrator was able to plant his malicious software script because his privileged access was not revoked immediately upon his termination. An enforced policy of swiftly revoking the access of terminated individuals should be a standard policy of every company—and it's not that hard to implement. Fannie Mae simply got bogged down in bureaucracy and unnecessary procedures, delaying pulling the plug on the administrator's access until it was too late.

The Fannie Mae episode could have been easily averted had the organization created and enforced three simple policies:

- **Limit the rights of administrators**. Native Unix takes an "all access" approach to administrator permissions, violating the basic premise that every security manager knows: "Trust no one." Granting administrators everything they need to do their jobs, but nothing beyond that, brings a new level of order and common sense.
- **Shut down access quickly when necessary**. Some companies physically escort terminated employees and contractors off the premises, and although being walked off the property by a security guard is decidedly embarrassing, it is an unfortunate necessity. A single employee with a grudge can cause a lot of damage if left alone for even a few minutes, especially if he or she has access to and knowledge of the IT system. Sound policy must allow HR to terminate all computer access just prior to the employee receiving notice.
- **Track administrator activity**. Many organizations have a system to track what employees are doing, but that tracking often doesn't extend to the superusers. Existing technology can record keystrokes and observe actions in real time, create an audit trail and alert upper management that something is amiss before the damage is done. And many solutions can also save the session for forensics analysis and playback later.

Of course, a comprehensive answer to the problems of privileged access goes beyond a single solution; it involves a combination of enforceable policy and the right mix of broad enterprise solutions and specific technology solutions designed to satisfy compliance requirements and close the potential security holes created by the existence of multiple privileged access accounts.

Admins are busy, so convenience factors matter

Admins are overworked, which is why they tend to take shortcuts like writing privileged access passwords on paper and sharing them with one another. The idea of imposing a whole new protocol for privileged access will never get buy-in if it also imposes too many requirements that take extra time.

For instance, the largely manual and labor-intensive "firecall ID" scenario can break down very quickly, and basic sudo doesn't work beyond just a few servers. It's too much extra work for a group of people that are already trying to pack 12 hours of work into a 10-hour day.

Instead, management of privileged accounts must be automated, role-based, easy to use, and centralized across all systems with policies uniformly applied.

The idea of imposing a whole new protocol for privileged access will never get buy-in if it also imposes too many requirements that take extra time.

Dell's solution, Privilege Manager for Sudo, picks up where sudo leaves off, providing more granular control over policy, enhanced monitoring and the ability to manage delegation across multiple servers.

Granular access: adopting the least-privilege model

Instead of the universal access granted by privileged accounts, organizations need to be able to provide access on an as-needed basis, based on each individual's specific role. This is the principle of least privilege: provide access to only what is needed, when it is needed. This is not available in the operating system, but must be implemented with added privileged account technology and supported by policy.

Implications for regulatory compliance

Compliance issues have impacted even the smallest businesses hard. Regulatory compliance requires businesses across all industries to implement a secure environment that safeguards personal information and proves compliance with auditable records.

Regardless of the particular piece of legislation with which a business needs to comply, privileged access is at the forefront of the compliance paradigm. Most compliance issues can be addressed, however, through separation of duties within the privileged access domain, along with audit capabilities.

The sudo project

Native sudo

The open-source sudo project has gone a long way towards resolving privileged account challenges that many enterprises face. Sudo solves the immediate problem of admins accessing more than what they really need: it delegates authority and restricts access based on each person's role.

The free sudo project may be adequate in some circumstances. But for a larger enterprise with serious security requirements, it might not go far enough. The biggest limitation of sudo is that it is not possible to natively create a single policy and apply and manage it universally across all servers and networks.

Another limitation of sudo is that there is no audit trail and no visibility. Moreover, there is no centralized policy control, so management of the sudo environment is cumbersome and not standardized between servers. Sudo is widely used and a very common solution, but not a complete one for enterprise environments.

Plug-ins from Dell

Beginning with version 1.8 (February 2011), sudo architecture allows anybody to write plug-ins and add functionality to sudo. Dell Software is committed to improving the sudo platform, first by employing Todd Miller, the maintainer of sudo, to help keep the project alive and move it forward, and by offering a series of commercial enhancements. Dell's commercial solution, Privilege Manager for Sudo picks up where sudo leaves off, providing more granular control over policy, enhanced monitoring and the ability to manage delegation across multiple servers.

Dell's first commercially released sudo plug-in is a central policy server. In the past, one of sudo's limitations was that it had to be managed individually on every server on which sudo was installed, and there was no integration between servers. This led to a lot of redundancy and the need to rewrite identical policies for each server. Now, with Privilege Manager for Sudo, users can create policies from a single policy engine and push them out to everywhere they are needed.

The second sudo plug-in from Dell is a keystroke-logging module, which adds an extra layer of visibility, accountability and auditability.

Securing superuser access with Dell solutions

A set of products that work together to solve your privileged account management challenges

Dell Software's approach to privileged account and identity management is a set of independent products, that work together to solve the vexing problems associated with privileged accounts.

PPM replaces the laborious, manual process of the "firecall ID" with an appliance, making the process of password management automated, centralized and policy driven. Possession of the password can be set for a specific time or for a specific task, after which it is automatically revoked and changed.

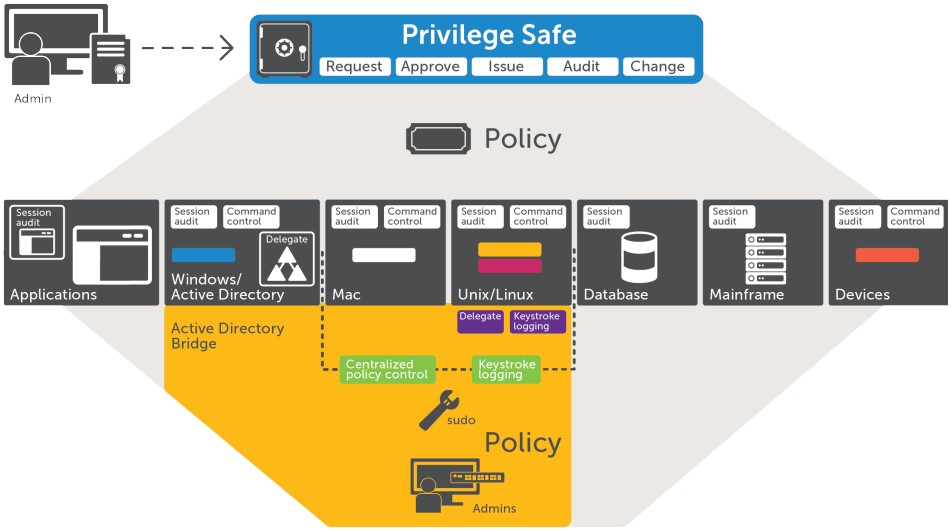


Figure 1. Dell Software security solutions include comprehensive offerings that address the privileged account management needs of even the most diverse and demanding enterprises.

Ease of use

Dell solutions deliver the advantages of a common standards-based approach, without the heavy requirements and administrative burden required by an all-encompassing, "big box" approach. With solutions from Dell, companies use only what they need, keeping costs down and eliminating unnecessary layers of administration.

A privilege safe enables centralized and policy-based release of privileged account credentials

One key function of Dell's advanced approach to account management is a "privilege safe." Privileged Password Manager (PPM) allows for centralized and policy-based release of privileged account credentials, without limitations from platforms, servers or devices—PPM works across the board on everything.

PPM is also designed to deal with the passwords that are typically hard-coded into applications. There may be dozens or hundreds of administrators who have, over time, learned those hard-coded passwords—an obvious security risk. Dell eliminates the need for hard-coded passwords; instead applications and databases are configured to make runtime calls to PPM. With this approach, nobody knows the application passwords, and the passwords can be changed rather than being locked into scripts, which is the real security and compliance concern.

Session management includes full keystroke logging and more

The ability to watch what people are doing is important to any system of checks and balances. Privileged Session Manager (PSM) from Dell supports full

keystroke logging with search capability. In addition to logging keystrokes and specific commands, PSM enables managers to watch over things as they happen on the screen and play back recorded sessions after the fact, whether the session was on Unix, Windows, Active Directory, Web applications, databases, devices or mainframes.

PSM provides an extra layer of accountability and visibility, including the ability to remotely kill a session or revoke access if needed. In addition, the chore of proving compliance or discovering the cause of trouble is bolstered by forensics-ready recording and playback of privileged access sessions.

Command control and privilege delegation

Privileged account management uses a system of delegation and control to limit what privileged users can and cannot do. Privileged Session Manager also provides whitelist capabilities that can specifically govern what any user can do. This ultra-secure approach enables a granular delegation of tasks and authority, giving each administrator the tools necessary to do his or her job, but nothing more. PSM works across many environments, including Unix, Linux, Windows and Web applications.

Another option available within Unix and Linux environments is a real-time, agent-based, granular delegation solution, Privilege Manager for Unix that runs on a server, providing both whitelist and blacklist capabilities and almost infinite control over policies and policy enforcement.

Either option provides absolute control over policy creation and enforcement. Not only can responsibility be delegated, it can be delegated based on role and time. For example, an administrator may be delegated specific privileges during the week, but if they have different tasks on the weekend, they can be assigned root access for that specific period of time or under very specific circumstances.

Appliance-based, host-based and agent-based options

A variety of delivery options are available to match every type of deployment need. Appliance-based solutions are extremely secure, easy to implement and easy to manage; just plug in the appliance and it's hacker proof. Host-based solutions are super secure, controllable and granular but slightly more expensive. Agent-based options deliver highly targeted, highly granular delegation for Unix, Linux and Active Directory.

Conclusion

The problems that arise from uncontrolled access to privileged accounts can result in multi-million dollar losses. Fortunately, powerful, cost-effective solutions are readily available to protect your business.

Dell Software's suite of privileged account management solutions give you the comprehensive accountability and granular access control that are missing from native operating systems, delivering a framework of least privilege so that administrators have access to what they need, but only what they need at the time they need it.

Dell's ultra-secure approach enables a granular delegation of tasks and authority, giving each administrator the tools necessary to do his or her job, but nothing more.

For More Information

About Dell

Dell Inc. (NASDAQ: DELL) listens to customers and delivers worldwide innovative technology, business solutions and services they trust and value. For more information, visit www.dell.com.

If you have any questions regarding your potential use of this material, contact:

Dell Software
5 Polaris Way
Aliso Viejo, CA 92656
www.dell.com
Refer to our Web site for regional and international office information.