

---

# FOXT.

Access Control Excellence

Fine-Grained Access Controls  
Management

*A Practical Guide*





*In order to control which servers users can access, including when, and how, you need to have a well-organized server network and processes in place defining what kinds of job roles should have access to what resources. This includes managing virtual servers so that users are not able to freely create virtual machines willy-nilly that fall outside of the access control framework.*

Fine-grained access controls are key to establishing secure access to enterprise resources and passing security and compliance audits. However, creating and implementing fine-grained access controls in enterprise server environments running a mixture of different operating systems requires careful planning and demands the right tools.

This paper offers a practical guide to implementing fine-grained access controls with ten aspects that are important to consider when planning an implementation of fine-grained access controls in a corporate server environment.

Top ten tips:

1. Organize your server environment.
2. Centralize user administration – one user, one account.
3. Get control of how users authenticate – tailor authentication to risk.
4. Make the most of existing security protocols (SSH).
5. Establish fine-grained access rules.
6. Protect the use of privileged accounts.
7. Establish and enforce enterprise password policies.
8. Audit-proof your systems.
9. Proactively monitor your servers.
10. Don't forget desktops, applications and other parts of your infrastructure.

## 1. Organize your server environment

*In order to control which servers users can access, including when, and how, you need to have a well-organized server network and processes in place defining what kinds of job roles should have access to what resources. This includes managing virtual servers so that users are not able to freely create virtual machines willy-nilly that fall outside of the access control framework.*

For example, a particular developer role may need access to UNIX, Windows, and virtual servers to do their work. To centrally manage all server access for this role, you need to be able to group these diverse

servers in one logical grouping which can then become a managed entity to which access can be granted or denied.

The exercise of going through your server environment and defining a structure is usually beneficial for the enterprise, as it firstly provides an inventory of machines in use and helps you recognize potential resource savings and synergies.

You will want to determine how best to organize your servers; For some organizations a function-based grouping will prove most appropriate, while others may take a geographical approach. Ideally, the solution you choose will support membership of multiple groups, so that an individual server can belong in, for example, a *Developer's* group and a *US* group.

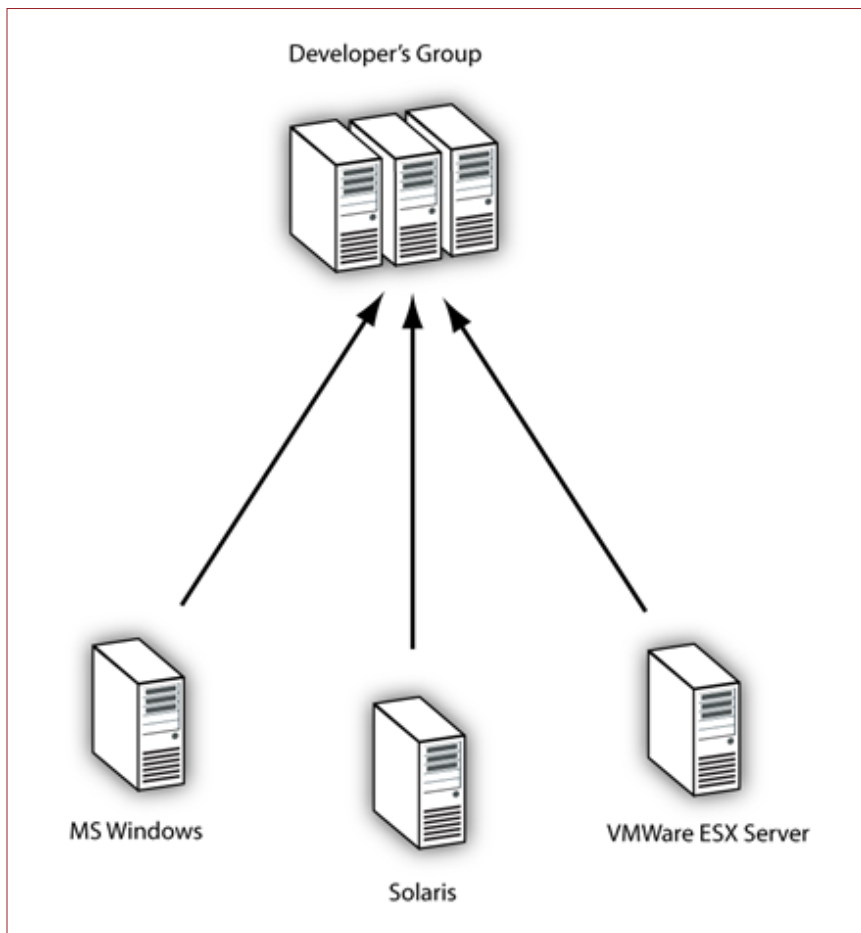


Figure 1: Organize your server environment.



*In order to provide accountability that can be traced back to individual users, you should minimize the use of functional accounts in your server environment. Strive to provision each physical user with a universal account that can provide the access that individual needs to various resources across the enterprise. This provides segregation of duties and traceability.*

While grouping servers requires you to create a meta-directory that can effectively manage the groups and access to the groups, once you have established this, it will act as the foundation for properly managing access to server-based enterprise resources. There are major advantages in utilizing one access control management platform across diverse operating system/server combinations, as it enables you to deploy one standard set of access control management policies.

## 2. Centralize user administration – one user, one account

*In order to provide accountability that can be traced back to individual users, you should minimize the use of functional accounts in your server environment. Strive to provision each physical user with a universal account that can provide the access that individual needs to various resources across the enterprise. This provides segregation of duties and traceability.*

Once your servers have been effectively organized into groups, the next step in providing fine-grained access controls is to consolidate user accounts into one account per user. There is little point in applying access controls for functional accounts if these are shared and the actions performed as those accounts cannot be traced back to a single user.

Meanwhile, in the world of business, user roles and organizations can change quickly and dynamically, with mergers and acquisitions providing real-life management challenges, so having robust mechanisms to deal with this can provide a concrete competitive advantage.

While most organizations are using some kind of user directory tool, many are finding that tools such as NIS/NIS+ and, increasingly, LDAP are being targeted by auditors as inherently lacking in security. There is little in-built control over management of user accounts in these tools, and an LDAP administrator can unwittingly be given unrestricted access to the user directories. Therefore, look for solutions that deliver secure consolidation of user accounts managed in a traceable way.

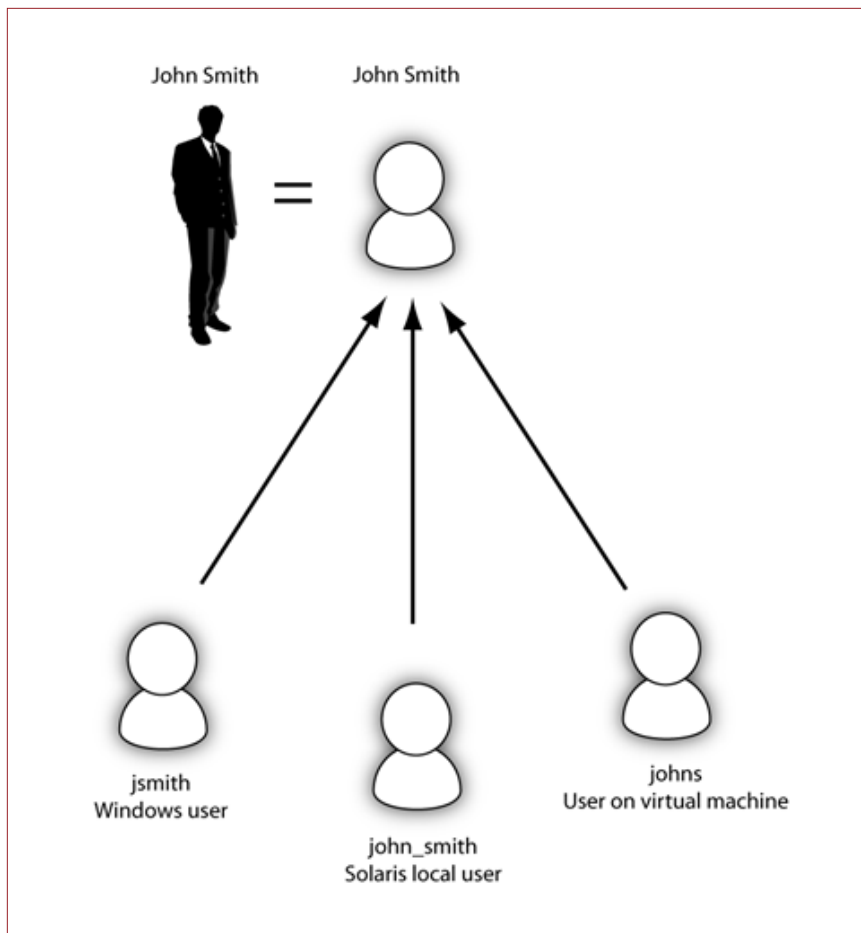


Figure 2: Centralize user administration.

Look also for a solution that can then allow you to group those consolidated user accounts into roles to make it easier to implement your access controls. This means that changes in access can be managed on a role level, and you can change the access parameters for hundreds of users in a single operation.

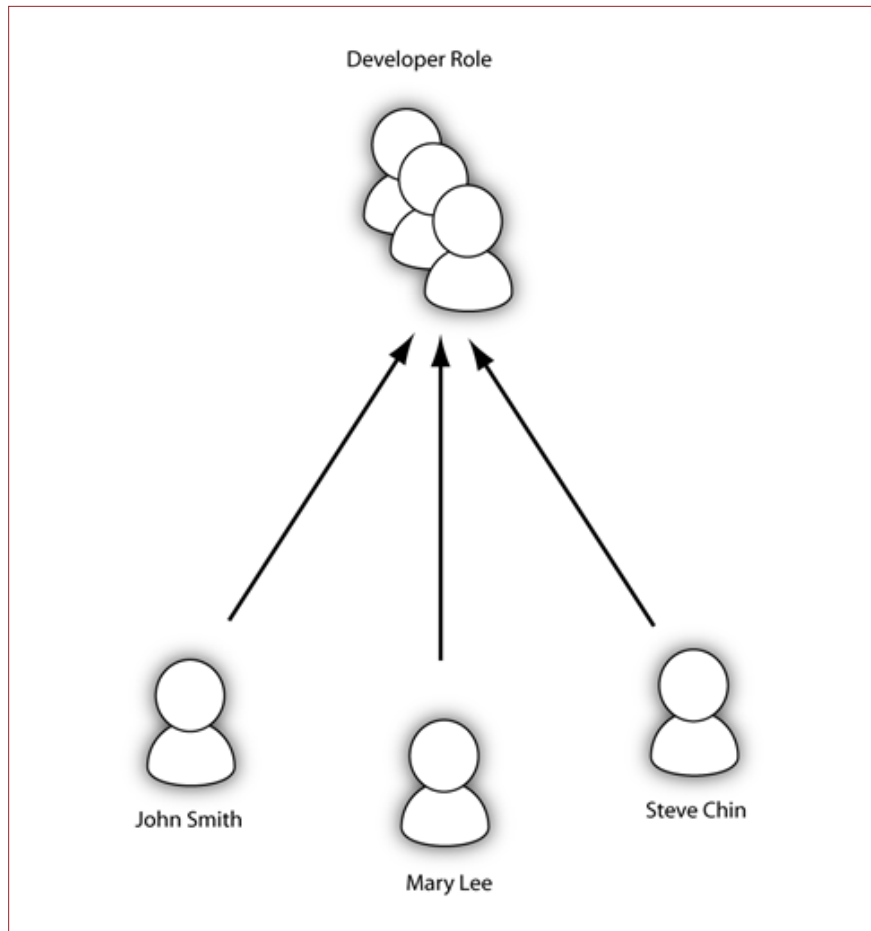


Figure 3: Group consolidated users into roles.

There are a number of advantages associated with consolidating user accounts:

- New users can be provisioned into the correct roles across the network of diverse servers in minutes,
- User access to resources can be blocked with one mouse click if required.
- It also becomes easier to trace and deal with orphaned accounts.

Look for a solution that will work with your existing infrastructure, not against it. Features such as LDAP and NIS integration can ease the headaches of implementation and allow your access control solution

to fit smoothly into your existing network infrastructure, and simplifies consolidating user populations as organizations change, merge and expand.

### 3. Establish fine-grained access rules

*You need a solution that can help you define access down to the service level, rather than the host or host group level. Only service level rules can truly limit user activities on a server to what they need to do their job. The ideal rules will support roles-based access control and be easily managed from a central point.*

There are many solutions on the market that will allow you to define user access to a particular server or group of servers. However by definition “fine-grained” access should allow you to define user access at a service level on a particular server at a particular time of day.

Once you have organized the server environment, established traceable one-to-one relationships between physical users and user accounts, and grouped users into roles you are then in a position to begin defining access rules for those users or roles. Ideally you will be able to work with rules securely from a central point such as a web-based interface.

It is advantageous to be able to specify the right kind of authentication for each access rule, and be able to add or remove access for users or roles instantly. Also look for solutions that can provide access rules for different target platforms (such as UNIX, Linux, Microsoft and virtual operating systems) that can be granted to the same user.

*You need a solution that can help you define access down to the service level, rather than the host or host group level. Only service level rules can truly limit user activities on a server to what they need to do their job. The ideal rules will support roles-based access control and be easily managed from a central point.*

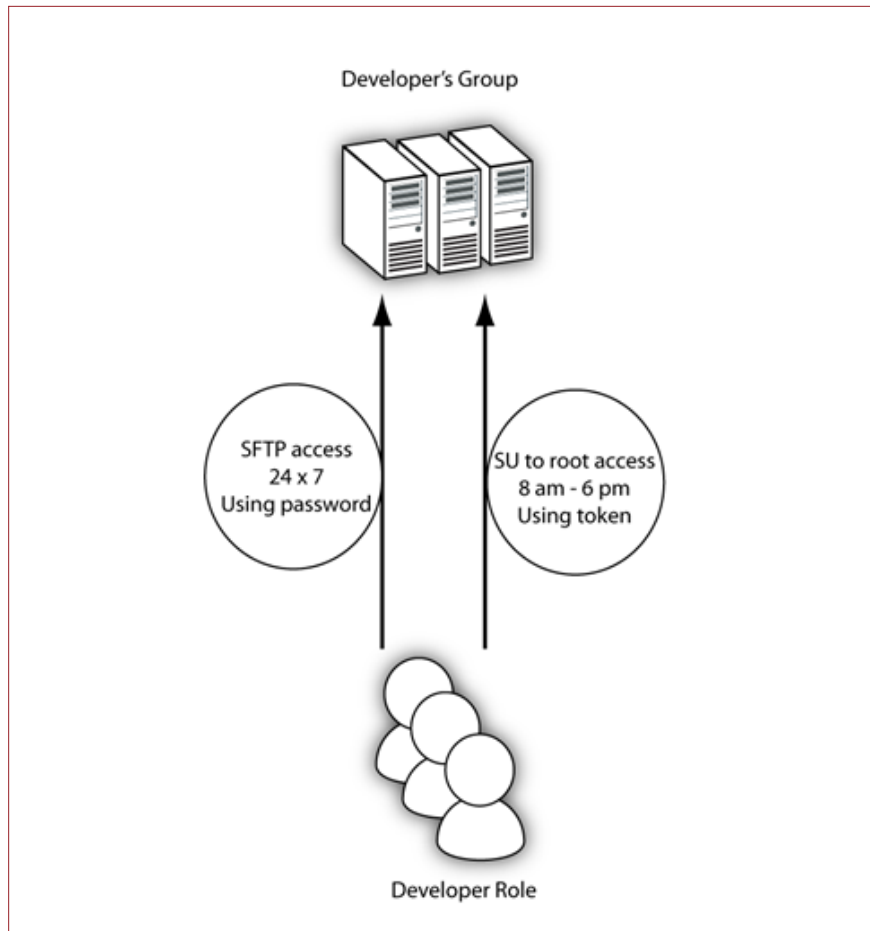


Figure 4: Create fine-grained access controls.

The wider the range of services and protocols the solution supports, the more fine-grained access controls you will be able to implement. Ideally the solution will work with the native security models on the target systems, supporting access controls to UNIX services on UNIX servers and Microsoft services on Microsoft Windows servers, working in harmony with the operating systems rather than imposing a foreign security model on them.

#### 4. Get control of how users authenticate – tailor authentication to risk

*Most organizations are still using insecure forms of password authentication, and sharing passwords between users. Shared passwords provide no traceability. At the same time, introducing*

*blanket strong authentication can be more of a security overhead than is needed, and may not be appropriate for lower-risk resources.*

The fact that users have a tendency to share passwords for sensitive and functional accounts just to “get the job done” has highlighted the need for stronger forms of authentication so organizations can be sure that authorized users are accessing their servers.

There are a number of authentication offerings on the market, giving organizations a wealth of alternatives to password authentication. One-time password tokens, USB tokens and smart cards all offer stronger authentication than standard passwords, but can be expensive and management-intensive to implement.

One way to cut the cost of authentication investments while protecting corporate assets is to target strong authentication to particular servers and roles that bring a higher level of risk. This requires flexibility in your authentication solution and also requires that you have been able to establish the “one user one account” principal as discussed in the previous step.

Consider the group of developers discussed above: You may want the developer to be able to access, for example, a time-reporting system using password authentication, but to use a smart card when accessing a sensitive code repository. Targeting authentication methods in this way allows you to avoid using blanket strong authentication.

Another aspect that should be taken into consideration is how many different forms of authentication your access control solution supports, giving you flexibility in your access control strategies moving forward. For example, if you are using LDAP for user directory management, does it support LDAP authentication? If you are using a PKI infrastructure, does it support certificate authentication? Does it support UNIX-to-UNIX single sign-on, allowing users to move between a defined set of servers without repeatedly having to re-authenticate?

To give you the option of implementing flexible authentication, your chosen solution should include a mechanism for easily adding and removing different authenticators from users.

*Most organizations are still using insecure forms of password authentication, and sharing passwords between users. Shared passwords provide no traceability. At the same time, introducing blanket strong authentication can be more of a security overhead than is needed, and may not be appropriate for lower-risk resources.*



*Many organizations do not want to deploy strong authentication across the enterprise and are therefore reliant on password authentication to some degree. Make sure you are using robust password policies across the network including specifying rules for password formats and enforcing regular password changes for accounts.*

Getting the mix of authentication settings right can minimize the administrative burden and cost of authentication infrastructure while ensuring that your higher-risk assets are appropriately protected.

## 5. Establish and enforce enterprise password policies

*Many organizations do not want to deploy strong authentication across the enterprise and are therefore reliant on password authentication to some degree. Make sure you are using robust password policies across the network including specifying rules for password formats and enforcing regular password changes for accounts.*

Some basic controls on user passwords in the enterprise can include forcing users to change their passwords after a certain period of time, and banning the reuse of passwords. However, if you are facing a large server environment running different operating systems with different methods of controlling passwords, even implementing such rudimentary controls can be a time-consuming job.

It is beneficial to be able to enforce such controls across a network of mixed servers running different operating systems. This is another benefit of achieving consolidated user accounts, as the password for that user account can be the same across the server environment.

Consider solutions that provide a robust password engine including the ability to generate long, random passwords that offer better security against “brute force” password attacks. Look for automated password synchronization across the server environment so users can log in on any server with the same password.

Password vaults are another method of enhancing password security and controlling how account passwords are used. Using some or all of these methods, you can ensure maximum security for accounts that are using password authentication rather than strong two-factor authentication.

## 6. Make the most of existing security protocols (SSH)

*There are a wide range of security protocols in the public domain, including open-source offerings such as SSH. While SSH has now been commercialized by a number of vendors, many organizations are reluctant to deploy these security protocols as they can bring with them a significant management overhead and lead organizations to change their business processes. Instead of making the most of existing security protocols to provide secure access to resources, therefore, there is a tendency to avoid deploying the likes of SSH because of management headaches.*

Having the right access to server-based resources is essential for your administrators to be able to do their jobs efficiently and effectively. SSH is one protocol designed to provide secure, encrypted access to servers. Originally an open source project, SSH is now available in a number of commercial varieties. However, the complexity of managing user and host keys across a large, diverse server environment can prove a real stumbling block and can in the worst case discourage organizations from making full use of SSH.

Consider solutions that can add a managed layer to SSH that fits in with the access control structure for your servers. Being able to automate deployment of SSH keys to hosts and users cuts the administrative overhead and makes the implementation of SSH in the enterprise much less daunting.

Another aspect to consider is whether your chosen SSH solution supports a full range of authentication mechanisms so you are not limited in deploying the secure protocol. Version 2 of the SSH protocol includes support for token and certificate authentication, which is preferable to password authentication for access to more sensitive information assets.

*There are a wide range of security protocols in the public domain, including open-source offerings such as SSH. While SSH has now been commercialized by a number of vendors, many organizations are reluctant to deploy these security protocols as they can bring with them a significant management overhead and lead organizations to change their business processes. Instead of making the most of existing security protocols to provide secure access to resources, therefore, there is a tendency to avoid deploying the likes of SSH because of management headaches.*



*Privileged accounts are especially important when it comes to implementing effective access controls. Any user who gets access to these accounts gets the power to do damage to systems and dispose of data. Organizations need a set of capabilities that enables them to handle privileged accounts without this getting in the way of the smooth running of day-to-day business processes.*

Finally, for true fine-grained access, it is essential that you can control SSH access on the sub-service level. This means for example that you can limit a specific user's access to the SFTP sub-service allowing them to securely transfer files, without granting them full SSH access whereby they can open an interactive shell and perform more operations.

## 7. Protect the use of privileged accounts

*Privileged accounts are especially important when it comes to implementing effective access controls. Any user who gets access to these accounts gets the power to do damage to systems and dispose of data. Organizations need a set of capabilities that enables them to handle privileged accounts without this getting in the way of the smooth running of day-to-day business processes.*

Sharing password information for privileged accounts has been highlighted as one of the main security and audit concerns in enterprises today. If you have a group of users sharing the root password for a UNIX server, for example, it is normally impossible to trace who did what. Given that root users in UNIX have very widespread powers, this is a dangerous state of affairs.

At the same time, many day to day data operations, such as system maintenance tasks and batch data transfers, can require privileged access, so you need a strategy that protects your data while ensuring the smooth day-to-day running of your business. Beware of implementing complex processes for privileged account management that actually hinder data operations.

The good news is there are a number of strategies you can adopt to protect and control the use of privileged accounts in your network that do not need to have a negative impact on day-to-day business.

Consider methods to avoid use of privileged account passwords, such as solutions that can relocate system administration to a fully-logged management console, that can allow users to perform privileged operations using their own account in a controlled way, and that can provide detailed monitoring of privileged account activities.

Explore ways of securely delegating management tasks to helpdesk personnel and implementing special controls for users su'ing to the root account on UNIX and Linux systems.

Password vaults that enable users to check out an ever-changing privileged password in a controlled manner can also be worth considering as part of your strategy to control the use of privileged accounts.

## 8. Audit-proof your systems

*Centralize auditing for hundreds of servers to help you more easily prepare for, and pass, security and compliance audits. Three crucial areas to consider are tracing effective user permissions, building a usable, comprehensive audit trail, and tracing root user and privileged account activities in the system.*

With a great increase in regulatory requirements over recent years, the amount of time and effort organizations are spending preparing for and undergoing audits has also increased. While become familiar with legislation and interacting with auditors has become a full time job for many, preparing to succeed in server audits comes down to answering three central questions: Who can do what on the servers? Who did what on the servers? What did privileged users do on the servers? Being able to answer these questions will give your organization a great chance of passing your security and compliance audits.

On the first point, consolidating physical user identities into one digital user account across all your servers is a great first step. Once a user only has one account across various mixed-operating system servers, you are spared the task of collating multiple digital identities before you can trace activities back to a physical individual. If you have associated your fine-grained access rules to these accounts, it is relatively simple to produce reports that map out user access permissions across the server environment.

*Centralize auditing for hundreds of servers to help you more easily prepare for, and pass, security and compliance audits. Three crucial areas to consider are tracing effective user permissions, building a usable, comprehensive audit trail, and tracing root user and privileged account activities in the system.*



Next, you need to produce evidence that the access rules you have implemented in the server environment are actually being enforced. This is done by means of an audit trail. However, consolidating and cross-referencing local audit logs from hundreds or thousands of servers running UNIX, Linux, Microsoft Windows and even virtual machines is an extremely difficult task. Look for a solution that can deliver a centralized audit log across the managed server environment with good reporting capabilities that will enable you to easily generate consolidated audit logs.

With privileged account activities warranting special attention from auditors given that privileged users can do more damage than ordinary users, make sure you are logging when users switch to privileged accounts. Normally this is difficult since, in UNIX and Linux systems particularly, root and privileged user actions are recorded but what user switched to root and actually carried out the actions is not logged. Based on the log files, it is impossible to establish which of the administrators did what. Furthermore, the root account includes privileges to actually tamper with the local log file itself, so wrongdoings can be covered up.

Look for a solution that can consolidate audit logs and provide special logging capabilities for privileged operations. As well as minimizing the use of privileged accounts, the solution should include more detailed logging capabilities for privileged operations, such as keystroke logging. In addition, it is important to consider what reporting options the solution offers to help with smooth preparation for audits in your server environment.

## 9. Proactively monitor your servers

*Establish routines for monitoring activity on your servers and making sure that access controls are working. File monitoring can help to ensure that sensitive files are not being tampered with, and integrity checking proactively scans systems for vulnerabilities, enabling you to stop breaches before they happen.*

Once you have established fine-grained access controls to your server environment, it is important to work proactively to ensure they are working properly. Find the right tools for scanning across the network that can provide appropriate alerts in the event of file tampering and access breaches. It never hurts to ensure an extra layer of security to make sure your access controls are working.

Look at solutions that include vulnerability scanning functionality that can be run at scheduled intervals. Examine how file permissions, file ownership and service configurations are monitored. Timeout functions that automatically log out inactive user sessions are another plus point that helps you keep on top of server security without diverting unreasonable amounts of administrative resources.

## 10. Don't forget desktops, applications and other parts of your infrastructure

*Take a holistic approach to access control and take advantage of your investments in data security tools across your data networks. Expand access controls management to desktops and applications and take the opportunity to standardize and consolidate controls in one common framework.*

Once you have established fine-grained access controls in your server environment, look at how the work you have done in this space can be extended to provide better controls in other parts of your IT infrastructure. Servers are of course only part of the overall picture

*Establish routines for monitoring activity on your servers and making sure that access controls are working. File monitoring can help to ensure that sensitive files are not being tampered with, and integrity checking proactively scans systems for vulnerabilities, enabling you to stop breaches before they happen.*



*Take a holistic approach to access control and take advantage of your investments in data security tools across your data networks. Expand access controls management to desktops and applications and take the opportunity to standardize and consolidate controls in one common framework.*

and, though an essential tool to running your business, their security also depends on what controls you have placed on desktop PCs and workstations, and applications.

Standardizing controls can provide significant long-term benefits in terms of cost and operational efficiency, and organizations who get it right the first time they establish a fine-grained access controls framework can reap considerable rewards over time.

Examine how your controls can be extended to these other elements of the IT infrastructure, and how you can use authentication and auditing controls applied in the server space to improve PC and application security. This will be easier if you have selected a server controls framework designed to expand to other enterprise IT infrastructure.

### Fine-grained access controls from FoxT BOKS Access Control For Servers

BoKS Access Control for Servers is an enterprise access control solution designed to provide fine-grained access controls to mixed server environments. The solution includes support for a wide range of UNIX, Linux, Microsoft and virtual operating systems.

The key features of BoKS Access Control for Servers are:

- Organize your mixed server environment by grouping servers into logical or geographical Host Groups
- Centralize user administration with one account per user across mixed server environments.
- Set flexible user authentication according to risk.
- Manage your SSH implementation with automated key management and fine-grained SSH access rules.
- Establish fine-grained, service level access rules to a large number of UNIX, Linux and Windows services.
- Protect the use of privileged accounts with a wide variety of access and auditing mechanisms including privileged command execution and keystroke logging functions.

- Establish and enforce enterprise password policies across mixed networks.
- Provide consolidated audit logs for mixed server environments and advanced log reporting.
- Proactively server monitoring with file monitoring, integrity checking and inactivity timeout monitoring.
- The BoKS framework can be easily extended to protect desktops and applications in the enterprise.

## Summary

Establishing fine-grained access controls can bring your organization a number of benefits, and help you achieve regulatory compliance, more easily pass audits, make better use of server resources, manage users more efficiently, and enhance security to protect your corporate data assets.

By following some straightforward principals, taking a practical approach to organizing your server environment, and choosing the right infrastructure solution to help you create and implement fine-grained access controls, you can realize all these benefits.

Copyright © 2008 FoxT. All rights reserved.

The document is provided for informational purposes only and the contents herein are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. The document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior permission.

FoxT logo is a trademark of FoxT, Inc. Other product and company names herein may be registered trademarks and trademarks of their respective owners.

