Global Information Assurance Certification Paper

NFS Security in Both Trusted and Untrusted Environments

GSEC Gold Certification

Author: Jakub Dlugolecki

Adviser: Jim Purcell

Accepted: October 21, 2007

Outline

1. Overview

This paper describes risks of using NFSv3 and NFSv4 in environments where performance is considered to be a more important factor than security. The paper also describes ways to mitigate those risks.

2. Introduction

Network File System protocol was created by Sun Microsystems in the 1980s as a file system for diskless clients. NFS provides remote access to shared file systems across networks. It was designed to be simple and efficient, not to be secure. Since 1980s the NFS protocol evolved. It was fit with many security enhancements. Unfortunately, security enhancements will never be as good as security foundations. Nowadays the NFS is used mainly in environments where performance is the main factor. A good example may be High Performance Linux clusters.

3. NFS security

3.1. NFS protocol overview

There are 3 versions of NFS protocols available. NFS Versions 2 & 3 are supported by all Linux 2.6 kernels. NFS Version 4 is supported by 2.6 Linux kernels but it also relies on Kerberos implementations in Linux distributions. It means that not all Linux distributions support NFSv4.

This paragraph will focus on NFSv3. NFSv3 has several more features than NFSv2. Unfortunately, none of those features is related to security. Both versions

are based on the same network security model.

Security capabilities provided by NFSv3:

- Server Authentication: none.
- Client Authentication: ability to define a list of authorized clients on a NFS server. Systems are listed by hostname or IP address. NFS server validates source IP address of incoming NFS requests.
- User Authentication: NFS relies on authentication methods provided by RPC protocol:
  - AUTH_NONE - no authentication
  - AUTH_UNIX - the NFS server implicitly trusts in UIDs and GIDs presented by a client. This is a most widely used authentication method.
  - AUTH_KERBEROS - Kerberos authentication. Not supported by most popular Linux distributions.
  - AUTH_DES - uses a combination of secret key and public key cryptography. Not supported by Linux.
- Data Integrity: NFS allows use of TCP or UDP protocols. In case of TCP, NFS data integrity is provided on a TCP protocol level. In case of UDP, NFS requires acknowledgements for every RPC command. NFS does not define what to do in case a packet with a RPC command acknowledgement is lost (commands on a filesystem may be executed twice).
- Data Confidentiality: Packets transmitted by NFS protocol are not

encrypted.

3.2. NFSv4 security improvements

While NFS Version 2 and 3 were designed to work in LAN network environments, NFS Version 4 was designed to work on the Internet. One of the main factors to create NFS Version 4 was to provide strong security, with negotiation built into the protocol. The two main security improvements in V4 are mandatory Kerberos support and improvements in username/UID management.

NFSv4 implements Generic Security Services API (GSS-API), called RPSEC_GSS. NFSv4 implementations must have two security flavors implemented in order to be compliant with NFSv4 standards:

- Kerberos Version 5 - suitable for intranet and local networks.
- LIPKEY - The Low Infrastructure Public Key (LIPKEY) system provides an SSL-like model and equivalent security for use on the Internet.

Security capabilities provided by NFSv4:

- Server Authentication: provided by LIPKEY. The client authenticates the server by comparing the latter's certificate with a list of trusted Certification Authorities (Pawlowski, 2005).
- Client Authentication: the same as for NFSv3.
- User Authentication: Kerberos Authentication
- Data Integrity: TCP protocol is mandatory for NFSv4. Moreover, RPSEC_GSS is capable of performing integrity checksums of entire body of NFSv4 call.

- Data Confidentiality: RPCSEC_GSS provides encryption of NFSv4 traffic. It means that NFSv4 provides encryption in transit. Kerberos 5 is most widely deployed RPCSEC_GSS security provider (Stern,2001). Kerberos employs strong data encryption algorithms like AES, 3DES.

NFSv4 support was added to major Enterprise Linux Distributions not very long time ago. It is considered to be "ready for enterprise validation". However, NFSv3 is still widely used mainly in order to provide backward compatibility for older UNIX versions.

4. Risks related to NFS (Risk Assessment).

4.1. Risk assessment scope.

Risk has many different components: assets, threats, vulnerabilities, safeguards, consequences and likelihood. This risk assessment will focus on data stored on NFS servers and technical consequences if this data is compromised. The risk assessment is based on an assumption that data availability is a bit more important for a company than data confidentiality and data integrity.

4.2. Environment description.

I would like to show in this paper how risks differ depending on environments in which NFS is used. Risk assessment will be done for two kinds of environment: "secure data centers" and "untrusted environments". For both those environments, NFS related threats and vulnerabilities are similar. Only the likelihood of threats and threat vector details differ.

4.2.1. Secure data centers - environment description.

All hosts which have an access to NFS shares are located in a one data center together with NFS servers. All hosts belong to the same LAN network. The LAN provides connection to Internet via properly configured firewall. The physical network topology allows connecting hosts to the LAN network only inside the data center. Physical access to the data center is protected.

All hosts inside the data center are managed/operated by the organization sysadmins. Hosts have all brands of Linux distributions, not all of them have latest security patches installed. Employees have regular UNIX accounts on hosts in data center. Users connect to servers inside the data center using SSH protocol.

4.2.2. Untrusted environments – environment description.

NFS servers are located in secure data centers but they export NFS shares to hosts outside the data center LAN network. NFS shares are exported only to hosts inside the organization. Client hosts are managed by organization sysadmins, but users have physical access to those hosts. There is an unlimited access to organization's WAN network from inside the organization.

"Untrusted environments" term is used because NFS clients cannot be trusted (anyone in the organization can run LinuxCD and become a NFS client). This environment differs from the Internet mainly because NFS servers can be trusted and because Kerberos infrastructure is in place.

4.3. Threats to Confidentiality, Integrity and Availability of data stored on NFS servers.

Data theft: there are many ways which can be used to steal data stored on NFS servers. Two most likely to happen are:

- Data theft in transit - attacker sniffs the network traffic and this way obtains the data which is being transferred. The impact is medium because an attacker is only able to obtain the data which is transferred (not all NFS shares).

- Remote data theft from NFS server - using a fake NFS client an attacker can connect to the NFS server and have an access to all NFS data on servers. For NFSv3 the impact is high, because most implementations of NFSv3 provide only AUTH_UNIX authentication which explicitly trusts to all requests provided by a client. It means that if NFSv3 is used, an attacker can get access to all data stored on NFS servers. The impact is much less severe if NFSv4 is used. When NFSv4 is used, even if an attacker manages to connect to NFS server, one has to provide valid Kerberos ticket to obtain a user's data.

Unauthorized data modification/removal: there are many threat vectors which may lead to unauthorized data modification. For example, an internal employee may exploit physical access protection vulnerabilities and physically access NFS servers. However, it is unlikely to happen. Two most likely threat vectors are:

- Data modification in transit: attacker injects modified NFS packets into network traffic. The impact is low because an attacker is only able to modify the data which is being transferred. The impact may be a little higher if root executable files are stored on NFS servers (this way an attacker would be able to modify scripts/executable files which are executed by root on NFS clients).

- Unauthorized remote access to NFS servers: the same threat vector is used as for "remote data theft from NFS server". The impact is high for NFSv3, since all data on NFS server can be modified/removed.

Interruption of operation due to network traffic instability: on high load networks computing environment may be affected by serious slowdown of NFS operations. The impact for the environment is medium.

Data Confidentiality/Availability/Integrity attacks using vulnerable NFS client/server implementations: NFS implementations have vulnerabilities as any other software has. The impact if those vulnerabilities are discovered and used would be high. The reason why this threat is described in this risk assessment is because NFSv3 and NFSv4 will be compared. Maturity of those protocols implementations differ so likelihood of discovering software vulnerabilities in NFSv3 and NFSv4 also differ.

4.4. NFSv3, NFSv4 security measures - secure data centers.

Threats to NFS data (in secure data centers).			
Potential threat	Probability (NFSv3)	Probability (NFSv4)	Impact
Data theft in transit.	LOW	LOW	MEDIUM
Remote data theft from NFS server.	MEDIUM	LOW	HIGH
Data modification in transit.	LOW	LOW	MEDIUM
Unauthorized remote access to NFS servers (data modification/removal)	MEDIUM	LOW	HIGH
Interruption of operation due to network traffic instability.	MEDIUM	LOW	LOW
Data Confidentiality/Availability/Integrity attacks using vulnerable NFS client/server implementations	LOW	MEDIUM	HIGH

4.4.1. Protection against data theft.

Most implementations of NFSv3 provide only AUTH_UNIX client authentication. It means that NFSv3 server trusts to all requests of an authorized client hosts. This attack vector does not require any special skills or tools. That's why NFSv3 is considered to be as secure as the weakest NFS client in the environment. NFSv3 also does not provide any transit encryption.

NFSv4 provides security improvements for both described threat vectors. Even

if an NFSv4 client host is compromised, an attacker has to provide active Kerberos ticket in order to get NFS data. NFSv4 also provides traffic encryption.

Secure data centers infrastructure provides additional security measures against data theft:

- Network topology makes it difficult to sniff the traffic (physical access to the data center is required).
- Client hosts are located in a data center. It makes it difficult to mount NFS shares using fake NFS client. An attacker would have to compromise at least one host inside the data center in order to mount NFS shares.

Likelihood of data theft for NFSv3 would be high, but security measures provided by data center infrastructure limit this likelihood to medium. Likelihood of data theft for NFSv4 is low both for hosts located in data center and outside the data center.

4.4.2. Protection against data modification.

Security measures for threat vectors related to unauthorized data modification are similar to security measures provided against data theft. NFSv3 does not provide data integrity checks in transit. It can be configured to use UDP or TCP as the transport protocol (UDP is a default option for most implementations). Technically, it is much easier to inject spoofed traffic to UDP transfer than into TCP connection. Likelihood of data modification in transit for

NFSv3 is low, because secure data center network infrastructure provides additional protection. Likelihood of data modification using fake NFS client is higher. The main reason is that for both attack vectors, an attacker would have to compromise one host inside the data center. Once the host is compromised an attacker can either try to spoof the network traffic or just mount NFS shares and access them. The second choice is much easier to accomplish and provides more privileges (all shares are accessible) than the first choice.

NFSv4 have built in security measures to assure data integrity. The NFSv4 protocol requires computing and checking integrity checksums of all NFSv4 packets. Additionally, the NFSv4 traffic is encrypted. Likelihood of data modification in transit for NFSv4 is low. Likelihood of data modification of NFSv4 using fake NFS clients is also low, because NFSv4 do not trust NFS client requests and require user Kerberos tickets in order to grant access to NFS data.

4.4.3. Protection against interruption of operation caused by network instability.

NFSv3 is able to use UDP or TCP as a transport protocol. In high load networks, TCP protocol to transport NFS traffic is more reliable. Since for most implementations of NFSv3 the default transport protocol is UDP, the likelihood of interruption of operation caused by network instability for NFSv3 is medium. NFSv4 is able to use TCP protocol only. The likelihood of interruption of operation for NFSv4 is low.

4.4.4. Protection against vulnerabilities in NFS implementations.

NFS server software can be vulnerable both for remote and local attacks. NFS client software can be vulnerable mostly for local attacks (since it does not listen on any network interfaces). Data center network topology protects against remote attacks to NFS server from the network outside the data center. This limits the likelihood of vulnerability related threats.

NFSv3 implementations are far more mature than NFSv4 implementations. Moreover, NFSv4 protocol is more complex than NFSv3. It means that it is more likely to find vulnerabilities in NFSv4 implementations than in NFSv3 implementations. Likelihood of threat of exploiting vulnerabilities in NFSv3 implementation in data centers is low. Likelihood of threat of exploiting vulnerabilities in NFSv4 implementations is medium.

4.5. NFSv3, NFSv4 security measures - untrusted environments.

Threats to NFS data (in untrusted environments).			
Potential threat	Probability (NFSv3)	Probability (NFSv4)	Impact
Data theft in transit.	HIGH	LOW	MEDIUM
Remote data theft from NFS server.	HIGH	LOW	HIGH

Data modification in transit.	HIGH	LOW	MEDIUM
Unauthorized remote access to NFS servers (data modification/removal)	HIGH	LOW	HIGH
Interruption of operation due to network traffic instability.	MEDIUM	LOW	LOW
Data Confidentiality/Availability/Integrity attacks using vulnerable NFS client/server implementations	LOW	MEDIUM	HIGH

4.5.1. Protection against data theft.

NFSv3 provides no protection against data theft in transit. It is not designed to work in untrusted environments. The likelihood of this threat vector for NFSv3 is high. The likelihood of unauthorized remote access to NFSv3 servers is also high. If users have administrative access to hosts to which NFSv3 data is exported, they can drop their privileges to any UID and copy all data from NFS servers.

NFSv4 was designed to work in Internet and it provides security controls which limit the likelihood of both those threat vectors to low. NFSv4 encrypts the data in transit. NFSv4 also uses Kerberos user authentication. So even if a NFS client is compromised, it needs to provide data owners Kerberos ticket in order to get the data.

4.5.2. Protection against data modification.

NFSv3 provides no protection against data modification. Both threat vectors are very likely to happen in NFSv3. Kerberos support in NFSv4 and integrity checksums of all NFS packets limit the likelihood to low for both threat vectors.

4.5.3. Protection against interruption of operation caused by network instability.

Most NFSv3 implementations use UDP as a default transport protocol. Likelihood of NFS operations slowdown for NFSv3 is medium because it's very sensitive to network traffic quality. The likelihood of NFS operations slowdown for NFSv4 is lower because it uses TCP protocol and NFSv4 provides mechanisms for data caching.

4.5.4. Protection against vulnerabilities in NFS implementations.

There is no protection against vulnerabilities in NFS implementations. As every host from company network can connect to NFS servers, a likelihood of exploiting remote vulnerabilities in a NFS server is higher in untrusted environments than in data centers. The likelihood for NFSv4 is higher than for NFSv3, because NFSv4 are far less mature and far more complex than NFSv3 implementations.

5. Risk mitigation.

5.1. Identified vulnerabilities – secure data centers.

NFSv3 is not as secure as NFSv4 is, but risks of using NFSv3 in secure data centers are still acceptable. Two most important vulnerabilities in NFSv3 are:

- Lack of secure user authentication support in many NFSv3 implementations. It means that a person who have root account on a client host, can access all data on NFS server.

- Lack of data encryption and data integrity checks in transit.

NFSv4 can be considered as a secure network filesystem solution for data centers. The only vulnerability discovered in this risk assessment is the insufficient maturity of the NFSv4 software. It means that there is a relatively higher likelihood of discovering vulnerabilities in NFSv4 and exploiting them with high impact to the environment.

5.2. Identified vulnerabilities – untrusted environments.

Vulnerabilities discovered for untrusted environments are the same as vulnerabilities discovered for secure data centers. However, due to lack of safeguards provided by secure data centers likelihood of exploiting those vulnerabilities is much higher. NFSv3 in untrusted environments does not provide controls to assure data confidentiality, integrity and availability. The NFSv4 protocol does not have any of NFSv3 vulnerabilities.

5.3. Mitigating vulnerabilities in secure data center environment.

The only way to mitigate risks related to potential software vulnerabilities in NFSv4 implementations is to agree with the NFS software provider on short SLAs.

There are two ways to mitigate the risk of unauthorized access to NFSv3 servers:

- Enable Kerberos authentication. In many cases it's technically too difficult to accomplish.
- Secure all hosts which have access to NFS servers. Keep them up to date.

The second solution requires fewer resources (hosts have to be kept secure even without NFS). However, it's very important that if this solution is chosen, NFS data will be as secure as the weakest host in a data center. It is very important to establish a process of reviewing host access lists on NFS servers on regular basis. Environment in data centers constantly changes and it's very important to keep security access lists up to date.

There are implementation specific guidelines for securing NFS available. The very good example of such document is available on the NetApp web library: "Security in NFS Storage Networks".

Data encryption and integrity checks in transit in NFSv3 can be provided by: IPSec implementation or by tunneling NFS traffic via SSH. Both those solutions may

make the environment less stable. The likelihood of someone sniffing or altering the data transferred inside a data center is low. As the likelihood is low, and data availability is more important than data confidentiality, recommendation of implementing those encryption solutions is questionable.

5.4. Mitigating vulnerabilities in untrusted environments.

NFSv3 was not designed to work in untrusted environments. Network traffic related vulnerabilities can be easily mitigated by implementing IPSec or SSH tunnels. However, the highest risk for NFSv3 in untrusted environments is an unauthorized remote access to NFS servers. The only way to mitigate this risk is to secure all hosts to which NFS shares are exported. In large companies this solution would require too many resources. Recommended solutions are:

- Switch to NFSv4. It is sometimes not possible due to wide range of NFS clients in a company.
- Disable NFSv3 for untrusted hosts and export this data using samba protocol. It may sound weird but the windows solution provides more security controls than the UNIX one.

6. Conclusions

NFSv3 does not provide sufficient level of data security. The main problem with NFSv3 is that most of its implementations do not provide secure user authentication. The next version of NFS protocol, NFSv4, can be considered as a

secure network file system.

It is highly recommended to use NFSv4 whenever possible. Migration to NFSv4 is not a straightforward task and requires a lot of IT resources. Sometimes it is even not possible because NFSv4 is not implemented yet in all UNIX and Linux platforms. Companies have to compare the value of data stored on NFS servers and the price of migration to NFSv4.

NFSv3 security can be highly improved by implementing safeguards like: secure network topology, keeping all NFS client hosts up to date, implementing physical access controls to all NFS client hosts. However, the NFSv3 will always remain as secure as the weakest NFS client host in the environment.

References.

1. Odhner, Chris (2005, February). Security in NFS Storage Networks. Retrieved September 1, 2007, from NetApp Documentations Web site: http://www.netapp.com/library/tr/3387.pdf

2. Garfinkel, S (1996). Practical UNIX & Internet Security. O'Reilly

3. Pawlowski, Biran (2005). The nfs version 4 protocol. Retrieved September 2, 2007, from NetApp Web site: http://www.netapp.com/library/tr/3085.pdf

4. NIST Organization, (1995, Oct). An Introduction to Computer Security: The NIST Handbook. An Introduction to Computer Security: The NIST Handbook, from http://csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf

5. Sun, H. T. (2006, June). KERBERIZED NFS IN A NETAPP STORAGE SYSTEM USING A UNIX-BASED KERBEROS AUTHENTICATION SERVER. from http://www.netapp.com/library/tr/3481.pdf

6. Berriman, Ellie (2006, November). Unified Windows® and UNIX® Authorization Using Microsoft® Active Directory LDAP as a Directory Store. from http://www.netapp.com/library/tr/3458.pdf

7. Stern, H (2001). Managing NFS and NIS, 2nd Edition.