Popular Protocols Demystified

ITSO-Security Engineering Branch

Introduction

This paper provides a high level overview of popular protocols with a focus on the security of each. This information is useful when deciding which protocol is best suited for the activity and its environment. Many of these protocols were developed years or decades before the network growth in the 1990s that resulted in an increased need for securing networked devices and encrypting transmission of data. The changing context for electronic communications led to the development of more secure protocols performing similar functions as their insecure predecessors. Many of the predecessors continue to exist and serve a purpose in limited cases, such as when networks are physically isolated and the associated data and services are not critical to operations. This has led to some confusion about the security aspects of the protocols. This paper clarifies security related questions for the following protocols:

- Telnet and Secure Shell (SSH)
- File Transfer Protocol (FTP) and Secure File Transfer Protocol (SFTP)
- Simple Network Management Protocol (SNMP) v1, v2, and v3

Telnet and SSH

Telnet was developed in 1969 before the need for security that began in the 1990s. The Telnet network protocol provides bidirectional unencrypted text-oriented communication between networked devices via an unsecured virtual terminal connection. Telnet, by design, has been and still is an unencrypted and insecure protocol. ITSO-SEB, along with other security experts, including the SANS Institute[1], discourages the use of Telnet[2] for the following reasons:

- Telnet, by default, does not encrypt any data transmitted, including authentication (such as passwords) and configuration information. Packet sniffing is easily performed by anyone who has access to a networked device where Telnet is used. Therefore, all data, including the password and any other potentially sensitive data, is easily acquired without authorization. The password can then be used to access other more secure devices and their sensitive data. If the password is associated with a privileged account (e.g., administrator), the span of security impact is even greater.
- Most implementations of Telnet have no means to prevent man-in-the-middle[3] attacks.
- Telnet, by default, tries to connect to the remote host using Port 23 but has the ability to connect to any port that has a valid listener. This capability allows it to be used to spoof or exploit valid protocols, thereby increasing the security risk to your entire network.
- Commonly used Telnet daemons contain numerous known vulnerabilities[4].

[1] SANS Institute InfoSec Reading Room – Auditing and Securing Multifunction Devices
[2] Telnet can be turned off via your /etc/inetd.conf file
[3] Man-in-the-middle attack defined at http://www.itsecurity.com/

Given the widespread use of Telnet and its stated limitations/concerns, the industry responded through the development of security enhancements, such as Transport Layer Security (TLS) and Simple Authentication and Security Layer (SASL). However, these enhancements are not supported by all Telnet applications and require both the server and the client to be using compatible versions of Telnet. Therefore, the standard SSH network protocol, developed in 1995, is favored over Telnet as it provides much of the same functionality plus compatibility while being significantly more secure.

As of this writing, there are two commonly used SSH protocols. As often happens, the initial protocol, SSH1, has been shown to contain security vulnerabilities that its successor, SSH2, does not. Therefore, adopters of SSH1 are encouraged to transition to using SSH2. SSH2 is a cryptographic network protocol that includes strong encryption of all data transmitted, including authentication (such as passwords) and configuration information. SSH2 also uses public key authentication to authenticate the identity of the remote device, if necessary. Using SSH2 instead of Telnet greatly reduces the security risk of remote device access and administration. In addition, explicitly disabling fallback to SSH1 eliminates unexpected use of the older and less secure SSH1 protocol.

Considering making the transition to SSH2? Having discussed the concerns with Telnet, its ease of use and ubiquitous nature does little to fully discourage its presence. Whether used intentionally or simply not disabled (just in case it is needed), well known Telnet vulnerabilities may provide an attacker ingress to the network. So, to get started with SSH2, you will need to:

- Disable Telnet as part of your standard hardening activities.
- Close Telnet's communication port 23 if there are no appliances or applications, for example older printers or legacy applications, that still use Telnet. Consider upgrading any appliances or applications that require Telnet.
- Acquire a Digital Certificate by contacting AOml_SSL_Cert_Admin@ao.uscourts.gov. The AO has a bulk purchasing agreement for Verisign certificates that reduces the cost and time for acquisition.
- Install OpenSSL and any packages and libraries required for your environment. Most installs will require restarting the Operating System and hence need to be scheduled during a maintenance window or otherwise planned.

FTP and SFTP

FTP was originally specified in 1971[5] and has been updated a number of times since. Similar to Telnet, FTP was developed before the need for security that began in the 1990s and hence, was not designed to be a secure protocol. Consequently, it has many critical security weaknesses[6]. FTP, by design, is an unencrypted and insecure protocol for transferring files between networked computers. ITSO-SEB, along with other security experts including the SANS Institute[7], discourages the use of FTP because the

[4] SANS FAQ regarding Telnet including a list of Telnet exploits

[5] RFC 114 FTP Specification

[6] Many of these security weaknesses are enumerated in RFC 2577 and include Spoofing, Username Insecurity, Port Stealing, Packet Sniffing, Bounce Attacks and Brute Force Attacks.

[7] SANS Institute InfoSec Reading Room – Securing FTP Authentication

protocol contains a number of mechanisms that can be used to compromise network security, including the following:

- FTP, by default, does not encrypt any transmitted data, including authentication (such as passwords) and configuration information. Packet sniffing is easily performed by anyone who has access to a networked device where FTP is used. Therefore, all data, including the password and any other potentially sensitive data, is easily acquired without authorization. The password can then be used to access other more secure devices and their sensitive data. If the password is associated with a privileged account, such as an administrator, the span of security impact is even greater.
- FTP is also vulnerable to brute force, bounce, man-in-the-middle, port stealing, and spoofing attacks.
- Anonymous FTP, as its name implies, allows anyone access to the FTP server.
- FTP cannot ensure the confidentiality, integrity and availability of the data being transferred.

The SFTP protocol, which uses the SSH network protocol, is favored over FTP as it provides much of the same functionality and is significantly more secure. SFTP is a cryptographic network protocol that includes strong encryption of all data transmitted, including authentication (such as passwords) and configuration information. SFTP leverages the SSH use of public key authentication to authenticate the identity of the remote device (if necessary). Using SFTP instead of FTP greatly reduces the security risk of remote device access and administration. At a minimum, anonymous FTP should be disabled and FTP only used when SFTP cannot otherwise be leveraged based on a comprehensive security risk analysis.

Considering making the transition to SFTP? Having discussed the concerns with FTP, its ease of use and ubiquitous nature does little to fully discourage its presence. Whether used intentionally or simply not disabled (just in case it is needed), well known FTP vulnerabilities may provide an attacker ingress to the network. So, to get started with SFTP, you will need to:

- Disable FTP and anonymous FTP as part of your standard hardening activities.
- Close the FTP communication port 21 if there are no legacy applications that still use FTP. Consider upgrading any application that requires FTP.
- SFTP operates using the SSH daemon (SSHD) on the server and so the daemon must be installed prior to using SFTP.
- Open communication Port 22 which is used by SFTP. This will facilitate closing the less secure FTP port 21 that is no longer needed with the transition to SFTP.
- A few technical points should be addressed when transitioning from FTP to SFTP
    - SFTP is not backward compatible with standard FTP servers, nor can a standard FTP client connect to the SSH daemon used for SFTP.
    - Since the SFTP command sets are different from FTP, any application or device that uses FTP programmatically may fail.

SNMP V1, V2, and V3

SNMP first appeared as RFCs 1065, 1066 and 1067 (now referred to as SNMPv1) in 1988[8] and has subsequently evolved to SNMPv2 and SNMPv3. SNMP was developed to provide network management and monitoring capabilities before the need for security that began in the 1990s. Therefore, SNMP v1 was not originally designed to be a secure protocol and has many critical security weaknesses[9] , which have been incrementally addressed through v2 and v3. Encryption was not added until the security focused updates with SNMPv3. Therefore, by design, all versions of SNMP prior to v3 are unencrypted and insecure protocols. Because SNMP agents expose networked device (routers, servers, hosts, printers, etc.) metadata, such as type and description, and allow control of the managed devices, this data and control needs to be secured.

ITSO-SEB, along with other security experts (including the SANS Institute[10]), recommends discontinuing the use of all versions of SNMP prior to v3 because the earlier protocols contain a number of mechanisms that can be used to compromise network security, including the following[11]:

- Earlier versions of SNMP (prior to v3), by default, do not encrypt any data transmitted, including authentication (such as community strings) and configuration information. Packet sniffing is easily performed by anyone who has access to a networked device where SNMP is used. Therefore, all data, including the community string and any other potentially sensitive data, is easily acquired without authorization.
- Pre-v3 SNMP is also vulnerable to denial-of-service attacks, service interruptions, and may allow an attacker to gain access to the affected device.

SNMPv3 is a cryptographic network protocol that includes strong encryption of all data transmitted, including authentication (such as community strings) and configuration information. Still, all three available SNMP versions are vulnerable to brute force and dictionary attacks for guessing the community strings, authentication strings, authentication keys, encryption strings or encryption keys. If SNMP is used over UDP, it is vulnerable to IP spoofing attacks as well.

Considering making the transition to SNMPv3? Whether used intentionally to support older network appliances or applications or simply not disabled (just in case it is needed), well known SNMPv1 and v2 vulnerabilities may provide an attacker ingress to the network. So, to get started with SNMPv3, you will need to:

- Provide training to all who will be using SNMPv3 as the terminology is somewhat different although the architecture concepts are similar across versions of SNMP. The additional security and administration features will also need to be learned. Those already familiar with SNMP can

[8] IETF Documents for full text of RFCs

[9] CERT SNMP Vulnerabilities FAQ

[10] SANS Institute Resources – Widespread SNMP Vulnerabilities

[11] Cert Advisory CA-2002-03 Multiple Vulnerabilities in Many Implementations of SNMP

start by reviewing the relevant RFC documents for SNMPv3 readily available on the Internet at no cost.[12]

- Evaluate and plan to resolve coexistence issues relating to the three versions of SNMP. These issues are covered in RFC 3416.
- Older network appliances or legacy management applications that might not support SNMPv3 need to be identified and evaluated for replacement.

Summary

In summary, there are many protocols available to “do the job”, but not all protocols are otherwise created equal. As a standard practice, disable all protocols that are not specifically needed. This will minimize the security risk to your network and its devices and data by reducing the potential avenues of attack, e.g., if an insecure protocol is disabled, then its vulnerabilities are not available for an attacker to leverage. For each protocol, evaluate its associated risk and decide whether or not the risk is acceptable to all stakeholders in its operational context. This will usually limit protocol use to the newer and more secure protocols as they reduce the risk of your data being compromised. A summary matrix of the risks for the protocols reviewed in this paper is provided below.

	Overall Security	Secure Authentication	Encrypted Communication	Known Susceptibility to Common Attacks										
				Packet Sniffing	Spoofing	Man in the Middle	Denial of Service	Service Interruptions	Device Control	Username Insecurity	Password Insecurity	Port Stealing	Bounce Attacks	Brute Force Attacks
Telnet	Poor	No	No	X	X	X	X	X	X	X	X	X	X	X
SSH-1	Poor	No	Yes		X	X	X	X	X	X	X			X
SSH-2	*Excellent*	Yes	Yes							X				X
FTP	Poor	No	No	X	X	X	X	X	X	X	X	X	X	X
SFTP	*Excellent*	Yes	Yes											X
SNMPv1	Poor	No	No	X	X	X	X	X	X	X	X			X
SNMPv2	Poor	No	No	X	X	X	X	X	X	X	X			X
SNMPv3	Good	No	Yes		X			X	X					X

Again, if at all possible, disable FTP and Telnet. If you are using SNMP to monitor or manage devices, consider disabling SNMPv1 and v2 and using v3 to leverage its authentication and encryption capabilities. While not discussed herein, also disable HTTP for management and use HTTPS instead. Finally, any protocols that are made available should be locked down to the minimum number of hosts or subnets that require access.

[12] A SNMPv3 White Paper and links to all related RFCs and related materials available online is consolidated at the SNMP Research International, Inc. website