matasano

NetApp MultiStore: An Independent Security Analysis

Prepared By:

Thomas Ptacek and Eric Monti, Matasano Security

EXECUTIVE SUMMARY

In early 2008, Matasano Security conducted an extensive security audit of the NetApp® Data ONTAP® operating system and the licensed software feature, MultiStore®. At the conclusion of our testing, we found that the Data ONTAP operating system exceeded our expectations for security. Matasano Security knows of no vulnerabilities in Data ONTAP or MultiStore that would compromise data integrity in MultiStore virtual storage resources, or the compromise of NetApp FAS Storage Systems.

SYNOPSIS

For several years, Matasano Security has worked with enterprises to assess and mitigate risks in storage networking technology. We began in 2005, with the discovery of the first published exploitable vulnerability in an iSCSI storage appliance. Since then, we have worked with clients and their vendors to uncover, mitigate, and resolve storage risks and vulnerabilities. Software security is what Matasano does, and storage security is a Matasano practice focus.

Early in 2008, our team was offered an opportunity to work directly with NetApp to research the security of their Data ONTAP operating system and its MultiStore software for secure partitioning of network and storage resources. With complete access to FAS Storage System technology, a direct line to NetApp technical resources, and free rein to inspect the functional areas of the Data ONTAP system that we felt were most exposed to attacks, we set about evaluating the security of a leading storage system.

This document describes our efforts and our findings. We constructed a test environment based on a threat model that pitted a FAS Storage System against a determined, skilled attacker, willing to invest the time to research and develop new vulnerabilities in a storage system, and given full access to a compromised subnet. At the conclusion of our testing, we found that the Data ONTAP operating system exceeded our expectations for security. We know of no vulnerability in Data ONTAP that would allow an attacker to use a FAS Storage System, even if they have a login to a portion of it, to compromise another subnet.

INTRODUCTION

About Matasano Security

Matasano Security is an independent evaluator of software security. We take software, appliances, firmware, and services and assess them for security flaws. Working with the vendors of the affected products, we help ensure that those vulnerabilities are fixed. Founded in 2005 by David Goldsmith, Jeremy Rauch and Thomas Ptacek, we are one of the leading firms in our field.

This project was conducted by Thomas Ptacek and Eric Monti. Both of us have prior experience with storage technology in general and with NetApp in particular.

About NetApp MultiStore

NetApp is the leading provider of networked storage for enterprises. NetApp's flagship product is the FAS Storage System, an appliance that provides access to large arrays of disk storage using storage protocols. FAS Storage Systems typically comprise the backbone of enterprise storage systems.

MultiStore is a licensed software feature of NetApp Data ONTAP that enables enterprises to deploy a single FAS Storage System and partition it into independent "virtual storage systems" (vFiler™ units), each serving up a specific set of storage assets, each administered separately.

MultiStore "virtualizes" storage and network resources by creating multiple IP endpoints for storage clients, differentiating clients based on which IP they connect to. Each vFiler unit has logically segregated storage, inaccessible to other vFiler units.

About This Assessment

In a “Penetration Test”, a team of security experts is given a target and instructed to simulate the actions of attackers in order to compromise it. Penetration tests are commonly conducted against networks and web applications in order to test the effectiveness of security mechanisms. Penetration testers usually rely on security scanning tools such as “Nessus”, or on libraries of exploits for well-known security flaws, in addition to common-sense testing for weak passwords and configuration errors.

Our test of the NetApp FAS Storage System was similar to a penetration test, but differed in several important ways. First, we were testing not for resistance to known vulnerabilities, but rather for the existence of new, previously unknown vulnerabilities. Furthermore, our testing was conducted inside of storage and management protocols, and in that environment, there are no commercially-available scanning tools to test for vulnerabilities. Next, we were testing the resilience of the FAS appliance itself, and not the security of a network deployment with firewalls and access control; in our test, the FAS was exposed directly to attackers. Finally, as we will describe in the next section, our hypothetical attacker was given unusual access to the target FAS Storage System.

As a result, where a conventional penetration test would involve running security tools and analyzing the results, our test instead involved multiple weeks of storage protocol research and software development. For each of the protocols we tested, we developed new testing tools to attempt to uncover vulnerabilities. Our hypothetical attacker was thus not simply a disgruntled employee, but rather a team of skilled researchers willing to invest weeks of time to identify and exploit any software vulnerability in the NetApp storage protocol stack.

The MultiStore Threat Model

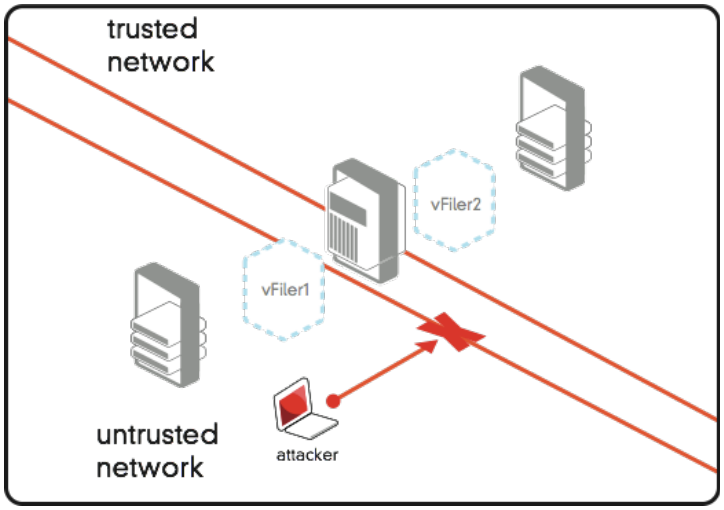
Assume a FAS Storage System is deployed with a MultiStore configuration in two environments. First, a high-security environment that stores customer protected information (PI). Second, a low-security environment for testing, QA, and development.

The high-security environment is tightly locked down to a small number of administrators, runs a minimal set of applications, and is fully patched. None of these steps have been taken in the low-security environment. Individual vFiler units support each environment, all sharing a single NetApp FAS Storage System.

Now assume the low-security environment is compromised by an attacker, who obtains full access to its vFiler unit. Can the MultiStore featureset of Data ONTAP protect the customer PI in the high-security environment?

In our threat model, attackers have the following assets:

- An administrative login to a vFiler unit on the target
- Configured storage and network access to CIFS, NFS, and iSCSI
- Access to the vFiler unit through SSH and HTTP management interfaces



Our attackers do not have:

- Access to the core FAS Storage System, above that provided by the MultiStore feature set.
- Direct network access to the high-security target environment
- The ability to intercept packets between the high-security vFiler unit and the servers in the high-security environment.

In an enterprise security environment, this is an atypically generous scenario for the attacker, who has been granted full access to a vFiler unit "for free". A typical attacker would not start out with a password to the Data ONTAP CLI, would not have direct access to a CIFS share on the system, or the ability to bring up an iSCSI session to a LUN on the vFiler unit. Here, we have assumed the worst-case scenario for an enterprise defending one internal environment from another.

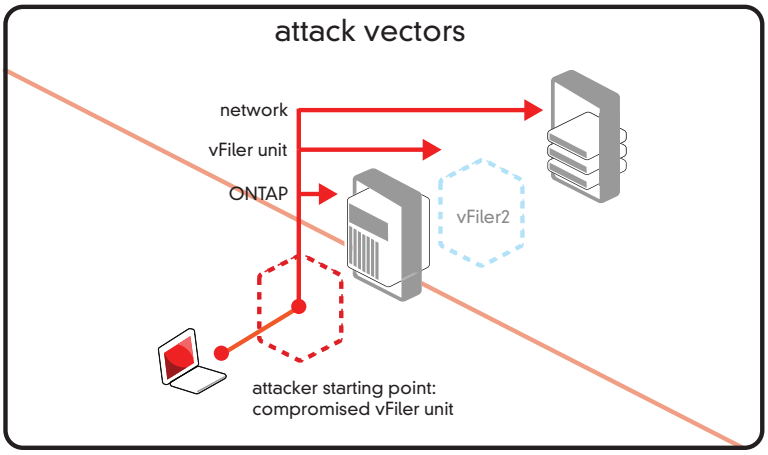
Our attackers win if they:

- Can access files or storage resources provisioned to the high-security vFiler unit from their compromised vFiler unit.
- Can run configuration commands in the high-security vFiler unit.
- Can run configuration commands or access the configuration registry of the core FAS Storage System.
- Can execute and run code in the kernel of Data ONTAP, across all vFiler units.
- Can use the network stack in Data ONTAP to route traffic from the compromised low-security environment to the high-security environment.
- Can use the network stack in Data ONTAP to route traffic from the compromised low-security environment to the high-security environment.

Attack Surface

Our attacker's objective is to exploit their access to a compromised vFiler unit to gain access to a high-security environment that uses another vFiler unit on the same NetApp FAS Storage System. To do that, the attacker considers the various features, protocols, and applications the compromised vFiler unit exposes. Together, this exposed functionality is called the "attack surface", and in our analysis it includes:

- **The CIFS File Sharing Protocol**: MultiStore vFiler units offer access to files using the Microsoft-style CIFS protocol. CIFS must provide access to storage resources allocated to the vFiler unit without exposing the assets of other vFiler units.
- **The NFS File Sharing Protocol**: MultiStore vFiler units offer access to files using the standard NFS protocol. NFS must provide access to NFS mount points configured for the vFiler unit without exposing files on other vFiler units.
- **The iSCSI Protocol**: MultiStore vFiler units offer access to remote disk storage using the iSCSI protocol. iSCSI must provide access to LUNs allocated to the vFiler unit without allowing iSCSI clients to access LUNs or disrupt services for other LUNs for other vFiler units.
- **The Data ONTAP Operating System**: Each MultiStore vFiler unit offers administrative access via the Data ONTAP command line interface (CLI). The CLI must be sufficient to configure the vFiler unit, but cannot allow a vFiler unit administrator to impact other vFiler units or the core storage system. The TCP/IP network stack in Data ONTAP must not allow attackers to route packets from an insecure environment to a secure environment.



THE CIFS FILE SHARING PROTOCOL

Overview

CIFS (also known as SMB) is the standard file sharing protocol used by Microsoft operating systems. CIFS is a complex protocol with a heritage dating back to NetBIOS LANs from the 1980's, with successive iterations adding layers of backwards compatibility, handshaking, and performance enhancements.

Adding to the complexity is the fact that CIFS is also a medium used by Microsoft environments to perform remote administration and management, meaning that many functions beyond simple file sharing are supported in the same protocol.

Finally, CIFS is notoriously under-documented; until recently, many aspects of the protocol were proprietary and discoverable only through reverse engineering.

Taken together, these issues have made it hard for security testers to scrub vulnerabilities out of the CIFS protocol stack. As a result, CIFS has been a notorious source of security vulnerabilities in Windows and Linux Samba server environments.

The MultiStore CIFS Threat Model

Regardless of whether the compromised FAS Storage System originally used CIFS, CIFS is a part of the attack surface of Data ONTAP. An attacker who has compromised a vFiler unit can simply enable CIFS, provided that CIFS has been licensed and enabled in Data ONTAP.

The following CIFS attack vectors require analysis for potential exposure in MultiStore environments:

1. Does the targeted high-security environment expose resources using the CIFS protocol, which could be accessed via a CIFS session to the low-security environment?

2. Is the CIFS protocol handling code in Data ONTAP vulnerable to software flaws that would corrupt the integrity or availability of the system?

3. Do the ancillary services exposed by CIFS, such as the computer "Browse" service (a network protocol that populates the "Network Neighborhood" feature of Windows operating systems) or the domain authentication and management subprotocols, themselves have vulnerabilities that impact the security of the Data ONTAP kernel?

4. Do any of the proprietary extensions to CIFS used by NetApp, such as the named pipes that support SnapDrive®, have vulnerabilities that impact the security of the Data ONTAP kernel? The same proprietary extensions could expose functionality that pertains to the kernel or other vFiler units.

The CIFS Attack Surface

In our testing, we considered the following areas of concern in the Data ONTAP CIFS functionality:

- Handshake, discovery, and other "pre-authentication" messages and services that CIFS makes available to network clients without passwords.

- CIFS messages delivered over NetBIOS/TCP (on port 139), NetBIOS/UDP, and direct SMB (on port 445).

- The NetBIOS name service, an analog to DNS customized for NetBIOS environments, including the handling of malformed NetBIOS names.

- Share and server-level access control.

- The various authentication mechanisms used by CIFS, including NTLM, the LanMan hash, and NTLMSSP.

- The basic CIFS file access messages.
- The "named pipe" subprotocol of CIFS that allows messages for ancillary services to be carried over CIFS channels.
- The proprietary network services made accessible by NetApp over CIFS services.

What We Tested

To test Data ONTAP CIFS code, Matasano programmed a custom, full-stack implementation of CIFS, including NetBIOS, NBNS, SMB, Transact messages and the named pipe subprotocol, and the NTLM/NT-Hash/LanMan/NTLMSSP authentication protocols.

In addition to allowing us to simulate malicious CIFS clients, our code also made every field in every CIFS message "fuzzable", meaning that would randomize its value or inject patterns of known vulnerabilities into the fields. CIFS is a complicated protocol; basic protocol messages can have over 100 fields, with intricate dependencies between the fields. Our code therefore allowed us to:

1. Generate legitimate CIFS traffic on the wire.
2. Capture the traffic using packet capture tools
3. Break the CIFS messages up into their constituent fields
4. Inject faults into all these fields

This methodology gave us strong coverage of the message handling code in Data ONTAP, and, by automating testing, eliminated some potential for human error and missed test cases.

During our CIFS testing, we analyzed the behavior of the Data ONTAP operating system and the subsystem that handled NetApp's proprietary "named pipe" CIFS extensions. Using the results of that work, along with our testing and debugging tools, we were able to discover the message types exchanged over those extensions.

Testing Results

At the conclusion of testing, no vulnerabilities were apparent to us that violated the security of Data ONTAP in our threat model. In particular:

- We know of no exploitable vulnerability that exposes shares or files from one vFiler unit to SMB sessions made to another vFiler unit.
- We know of no exploitable flaws in the CIFS message handling code in Data ONTAP that would corrupt the integrity or availability of a FAS Storage System.
- The ancillary services exposed by CIFS in Data ONTAP are minimal (unlike those of a general purpose Microsoft networking server), and we know of no exploitable vulnerabilities in them that would allow an attacker on one vFiler unit to impact the security of another vFiler unit.
- The proprietary NetApp extensions for SnapDrive and other NetApp services are minimal, and we know of no exploitable vulnerabilities in them that would allow an attacker to compromise the security of the Data ONTAP kernel.

THE NFS FILE SHARING PROTOCOL

Overview

NFS is the standard file sharing protocol used in UNIX® environments. Unlike CIFS, NFS was deliberately designed for server environments, and standardized through the Internet's IETF standards body. NFS is better documented and easier to test, with a variety of strong off-the-shelf test tools available.

Like CIFS, NFS also provides ancillary services besides basic file sharing. Unlike CIFS, these services are all related to filesystems, including file locking and access control. These services include the NFS file locking and status tracking system and the access-controlling "mount" daemon.

The MultiStore NFS Threat Model

Regardless of whether a compromised vFiler unit originally used NFS, NFS is part of the attack surface of the FAS Storage System. An attacker who has compromised a vFiler unit can simply enable NFS, provided that NFS has been licensed and enabled in Data ONTAP.

The following NFS attack vectors require analysis for potential exposure in MultiStore environments:

1. Does the targeted high-security environment support NFS? The attacker might be able to access by speaking NFS to the low-security vFiler unit.

2. Does Data ONTAP itself have NFS software flaws allowing the attacker to corrupt the integrity or availability of the entire system?

The NFS Attack Surface

In our testing, we considered the following areas of concern in NFS:

- Status messages, mount requests, and other messages in the "pre-authentication" functionality that clients can make to NFS servers without an NFS export entry.
- NFS file handles, unique numbers that identify files on the server, which allow clients to access files.
- The NFS mount service, which makes decisions about which clients should access files on the server.
- The NFS protocol encoding mechanism, called XDR, which defines the rules under which NFS messages are translated into binary and exchanged over the network.

What We Tested

To test Data ONTAP NFS code, Matasano programmed a custom, full-stack implementation of the NFS protocol, allowing us both to simulate a malicious NFS client and intercept and alter messages from an NFS client to the NFS services on a FAS Storage System.

On top of that code, we built a variety of NFS security testing tools, including:

- A fuzzing tool that injected faults in NFS messages and maliciously altered the formatting of the XDR-encoded RPC packets that carry those messages.
- Probes for the randomness and replayability of NFS file handles.
- Tools to spoof authentication attempts to the NFS mount service.

Testing Results

At the conclusion of testing, no vulnerabilities were apparent to us that violated the security of Data ONTAP in our threat model. In particular:

- We know of no exploitable vulnerability that would allow an attacker to enable NFS on a compromised vFiler unit and access files on another vFiler unit.
- We know of no exploitable software flaws in message encoding or handling that would allow an attacker to corrupt the integrity or availability of the Data ONTAP kernel.

THE ISCSI PROTOCOL

Overview

iSCSI is the de facto standard protocol for creating Storage Area Networks (SANs) over IP networks. SANs differ from classic file sharing protocols like CIFS and NFS in that they provide unmediated access to raw disks. Unlike an NFS or CIFS environment, iSCSI clients format their own filesystems, accessing the iSCSI server as if it was a disk.

The iSCSI protocol functions by translating "local" SCSI requests from an "initiator" into iSCSI protocol messages, which are carried over an IP network and executed at the iSCSI "target". iSCSI messages refer to "Logical Unit Numbers" (LUNs), a SCSI term which in a MultiStore environment is a synonym for "disk". The messages perform two basic functions:

- Handshaking and authenticating initiators and targets
- Carrying SCSI messages and responses (called "CDBs")

iSCSI is a relatively recent protocol and has not benefited from extensive security testing. On the other hand, the iSCSI protocol is relatively simple and is well documented.

The MultiStore iSCSI Threat Model

Regardless of whether a compromised vFiler unit originally used iSCSI, iSCSI is part of the attack surface of Data ONTAP. An attacker who has compromised a vFiler unit can simply enable iSCSI, provided that iSCSI has been licensed and enabled in Data ONTAP.

The following iSCSI attack vectors require analysis for potential exposure in MultiStore environments:

1. Does the targeted high-security environment use iSCSI LUNs, which could be accessible to iSCSI sessions made to the low-security environment?
2. Do the handshaking portions of the iSCSI protocol expose enough information to the attacker to authenticate successfully to the high-security environment?
3. Is the underlying SCSI command execution system in Data ONTAP vulnerable to software flaws that would corrupt the integrity or availability of the system?
4. Is the iSCSI protocol code in Data ONTAP vulnerable to software flaws that would corrupt the integrity or availability of the system?

The iSCSI Attack Surface

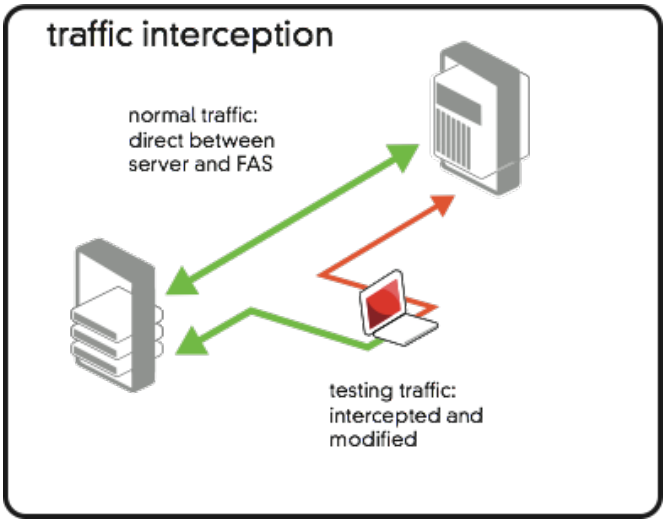
In our testing, we considered the following areas of concern in the iSCSI protocol:

- Handshake and "pre-authentication" negotiation messages that any network client can make to an iSCSI target without a password.
- Session management features (iSCSI supports a "bundling" feature that allows multiple TCP connections to be made on the same session).
- The CHAP authentication protocol that initiators use to log in to iSCSI.
- The iSCSI IQN target namespace, which initiators and targets use to identify each other.

- The iSCSI LUN namespace, which identifies individual disks on the target, and which is present both in the iSCSI header and in the individual SCSI CDBs, both of which need to be checked during iSCSI processing.

What We Tested

To test the Data ONTAP iSCSI code, Matasano programmed a custom, full-stack implementation of the iSCSI protocol, allowing us both to simulate a malicious iSCSI initiator and intercept and alter messages between a legitimate initiator and a FAS Storage System target.



On top of that code, we built a variety of iSCSI security testing tools, including:

- A fuzzing tool that injected faults in iSCSI handshaking and negotiation messages, to find parsing vulnerabilities, buffer overflows, and integer mishandling.
- A tool to test the CHAP authentication protocol used by iSCSI, which we used to execute test cases involving the CHAP state machine, and, in particular, whether CHAP messages for one target could be used to subvert a CHAP login for another target.
- A fuzzing tool that injected faults in SCSI CDBs, which are carried over iSCSI to SCSI drivers.

Testing Results

At the conclusion of testing, no vulnerabilities were apparent to us that violated the security of Data ONTAP in our threat model. In particular:

- We know of no exploitable vulnerability that would allow an iSCSI initiator to access LUNs allocated to other vFiler units, using handshaking messages, iSCSI Command messages, or SCSI CDBs.
- We know of no exploitable vulnerability that would allow an attacker to replay CHAP challenges from one vFiler unit to another, nor of any exploitable vulnerabilities in the execution of the CHAP protocols or their associated cryptographic operations.
- We know of no exploitable vulnerabilities in the Data ONTAP SCSI CDB handling code, nor any exploitable inconsistencies in handling information repeated inside CDBs also present in the iSCSI header.
- We know of no exploitable vulnerabilities in handshaking, negotiation, session management, iSCSI task management, or iSCSI message format handling in Data ONTAP iSCSI code.

THE DATA ONTAP OPERATING SYSTEM

Overview

Data ONTAP is the operating system that runs the FAS Storage System. All features of a NetApp FAS Storage System are implemented in the purpose-built Data ONTAP operating system.

Operators configure FAS Storage Systems by using the Data ONTAP CLI, which is made available over SSH. The CLI provides multiple privilege levels. In a MultiStore environment, a FAS Storage System's owners configure the entire storage system by connecting to a special vFiler unit, "vFiler0". vFiler unit administrators connect to the CLI on their particular vFiler unit. Data ONTAP also provides for administration over a web interface, and provides an SNMP MIB.

The MultiStore Data ONTAP Threat Model

Short of filtering IP packets, there is no way to disable access to the Data ONTAP CLI. By giving the attacker access to a compromised vFiler unit, that attacker also gets access to the Data ONTAP CLI.

The following attack vectors require analysis for potential exposure in MultiStore environments:

5. Can an attacker issue commands in the Data ONTAP CLI for their vFiler unit that would reconfigure the FAS Storage System proper or another vFiler unit?
6. Are debugging, diagnostic, and support commands accessible from a vFiler unit that would give an attacker privileged access to the Data ONTAP kernel?
7. Can an attacker access the Data ONTAP global configuration registry by disk, CLI command, SNMP, or web access, adding accounts or storage access rules to other vFiler units?
8. Is the Data ONTAP CLI vulnerable to software flaws that would corrupt the integrity or availability of the system?
9. Does the Data ONTAP web management interface expose vulnerabilities that would allow attackers to reconfigure the FAS Storage System without a password?
10. Is the Data ONTAP web management interface vulnerable to software flaws that would corrupt the integrity or availability of the system?
11. Does the IP stack of Data ONTAP route packets from one environment to another? This is a potentially serious problem because the FAS Storage System, in a MultiStore environment, will likely straddle multiple environments with different regulatory and security requirements.
12. Can the applications running on the Data ONTAP operating system, such as FTP or CIFS, be coerced into making outbound connections in other environments?

The Data ONTAP Attack Surface

In our testing, we considered the following areas of concern in the Data ONTAP operating system:

- The SSH server administrators connect to in order to issue CLI commands.
- The implementation of the individual CLI commands, including the availability of diagnostic and debugging commands from vFiler unit environments.
- The implementation of the CLI itself, including command parsing, tokenization, and metacharacters.
- The web management interface on the system.

What We Tested

Our testing of the Data ONTAP CLI was mostly manual and testcase-driven. Our methodology centered on binary code analysis of the Data ONTAP operating system to identify available commands, and attempts to execute those commands in various permutations against a vFiler unit. In the course of testing, we built a catalogue of commands available in vFiler units and commands available in the core CLI.

We conducted extensive testing of the Data ONTAP web management interface running both the HTTP and SSL-secured HTTPS protocol, using standard web application testing tools and techniques. In addition to domain-specific vulnerabilities, we assessed the resiliency of the web management interface against classic OWASP-style vulnerabilities, including metacharacter injection, cross-site scripting, cross-site request forgery, and forced browsing.

We also evaluated the various network services available on vFiler units, including storage protocols and the network commands exposed from the CLI, to ensure that a vFiler unit could not gain access to subnets connected to other vFiler units. Using IP stack fuzzing tools, we also ensured that the system could not be attacked using corrupted IP packet headers, IP options, or fragmentation.

Test Results

At the conclusion of testing, no vulnerabilities were apparent to us that violated the security of Data ONTAP in our threat model. In particular:

- We know of no way for an attacker to issue a CLI command against a vFiler unit and alter the configuration of another vFiler unit or the core system.
- No debugging or diagnostic commands we could uncover are made available to vFiler unit CLI sessions, and we know of no exploitable vulnerabilities in the CLI command handling code in the Data ONTAP CLI..
- We know of no exploitable vulnerability that exposes the NetApp configuration registry or its backing store to vFiler unit CLI or web sessions.
- We know of no way for an attacker to connect to the HTTP server of a vFiler unit and reconfigure the FAS Storage System, or of any way for an attacker to access the core configuration web interface without authenticating.
- We know of no exploitable vulnerabilities in Data ONTAP implementation of the HTTP protocol that corrupt the integrity or availability of the system.
- We know of no way for attackers to abuse the IP stack of a FAS Storage System to route packets to other environments, even at the application layer, including using FTP "bounce"-style attacks.`

CONCLUSION

As enterprises continuously strive to consolidate resources and infrastructure, storage administrators find themselves under increasing pressure to scale up and reuse centralized storage systems, instead of deploying individual systems for each application. This trend worries security teams, justifiably: when multiple applications depend on the same storage system, the cost of a security flaw in that system is amplified. It is an unacceptable violation of due care for a company to leave protected information vulnerable to storage system flaws exposed to untrusted networks and applications.

It's in these sensitive circumstances that we undertook our research into the features of NetApp MultiStore. If enterprises are going to rely on MultiStore, the feature has to work. It cannot be possible for an attacker to leverage access to a vFiler unit to take over the entire FAS Storage System, to access other vFiler unit resources, or to bridge their traffic from untrusted networks to protected networks. The extraordinary sensitivity of enterprise storage backbones merits extraordinary security assurance efforts.

As a result of the technology we were testing (storage protocols) and the circumstances of the test (mission-critical isolation features), our assessment committed extraordinary efforts to the attempt to break the security of the MultiStore feature. Rather than relying on well-known vulnerabilities and off-the-shelf testing tools, our test simulated the efforts of a well-funded, highly motivated team of attackers willing to research and develop new exploits for previously unknown vulnerabilities. Our testing spanned all the major storage protocols supported by NetApp MultiStore, including CIFS, NFS, and iSCSI.

At the conclusion of the test, our team can report that we know of no vulnerabilities that compromise the security model of the MultiStore feature:

- We know of no software flaws in the NetApp implementations of CIFS, NFS, or iSCSI that would allow attackers to exploit common C-code flaws like buffer overflows, integer overflows, or race conditions to execute code remotely in a FAS Storage System.
- We know of no architectural flaws in the storage protocols supported by MultiStore or their management interfaces, that would allow attackers to use access to a vFiler unit to reconfigure the FAS Storage System itself or any other associated vFiler units.
- We know of no protocol vulnerabilities in CIFS, NFS, or iSCSI that would allow an attacker to use a connection to their own vFiler unit to gain access to storage resources on other vFiler units, such as iSCSI LUNs or CIFS shares.
- We know of no vulnerabilities in the TCP/IP stack of the FAS Storage System that would allow attackers to bridge traffic from untrusted networks to trusted networks.

As any experienced security professional knows, it is always possible – even likely – that new vulnerabilities will be discovered in the future. There simply are no guarantees in information security. However, Matasano was given extraordinary access to NetApp resources and ample time to complete an in-depth assessment, and we conclude our engagement confident that MultiStore now exceeds the due care standards in place at our most stringent enterprise customers.