# Security Assessment Report
**DR120517**

## Siemens Enterprise Communications
## OPENSCAPE UNIFIED COMMUNICATIONS
## APPLICATION SERVER V7

*October 2012*

Miercom

www.miercom.com

# Table of Contents

# 1.0 Executive Summary

OpenScape Unified Communications Application Server v7 proved to be resilient and functional during multiple security attacks. The system was subjected to a battery of vulnerability analyses and scans without any additional security measures being deployed. OpenScape UC maintained performance even while DHCP, IPv4, ICMP, UDP and SSLv3 protocol mutations were directed to the Application Server.

The UC Application Server was subjected to thousands of exploits, along with 90,000 injection/fuzz/mutation attacks to the web application front end. Using only the security built into the OpenScape UC Server, none of the attacks were successful. Based on these results Siemens OpenScape Unified Communications Server v7 is Miercom Certified Secure.

A series of tests using Mu Studio Security, Metasploit, open source tools and Miercom proprietary scripts were performed to analyze the security of the OpenScape Unified Communications Application Server V7 (OpenScape UC Application) from Siemens Enterprise Communications. Tests included a complex set of exploits distributed by security tools and scripts to challenge the capabilities of the OpenScape systems and the OpenScape UC Application Server. The systems proved resilient through multiple series of tests.

The Unified Communications Application Server was tested without any additional security countermeasures employed other than what is provided in the system. The approach and methodology utilized in these tests are based on knowledge that Miercom, in collaboration with leading security experts, has collected from years of working in VoIP pre- and post-deployment site surveys and security assessments.

This document provides an overview of the more noteworthy exploit attempts that were conducted. In some test cases, specific details were intentionally omitted to avoid the use of this information to reverse-engineer exploits for VoIP products. The products used in testing were configured in accordance with the client's guidance, documented in their OpenScape Security Checklist that in effect enhances the resiliency of the systems. Siemens Enterprise Communications was afforded the opportunity to review initial findings, respond and repeat tests to ensure that potential vulnerabilities were addressed in the approved secure voice system.

Miercom tried to corrupt the OpenScape UC Application server by directing over 11,500 DHCP protocol mutations at the server. The UC Application server interface was challenged with over 31,000 IPv4, 42,900 ICMPv4, 6,400 UDP and 107,000 SSLv3 protocol mutations to disrupt or otherwise compromise its performance. Over 1,000 exploits were directed at the UC Application Server. All test results showed that the systems handled all attack vectors successfully with no severe faults. Performance and security of the servers remained normal. To gain internal privileged access, Miercom directed over 90,000 injections/fuzz/mutation attacks to the web application front end. None of these attacks were successful; the UC Application deflected all attempts to penetrate the system.

**Key Findings and Conclusions**

- DoS attempts directed against the Unified Communications Application Server were blocked by the internal firewalls.
- OpenScape system blocked all attempted exploits, while preserving the system's normal operation and the UC Application continued to operate.
- Over 100,000 protocol mutations were attempted against the UC Application Server, preserving normal system operation

An overview of some of the testing performed that is not confidential is detailed in the following sections. We were impressed with the performance of the OpenScape systems with the Unified Communications Server functioning while being subjected to malicious exploits and attacks. Miercom is pleased to present the Certified Secure Award to OpenScape Unified Communications Application Server V7.

Rob Smithers
CEO
Miercom

# 2.0 Introduction

Siemens Enterprise Communications engaged Miercom to perform a security assessment on OpenScape Unified Communications Application Server. OpenScape UC Application Server is a unified communications server that provides individual users a portal to instant messaging and presence status, a single interface to conferencing services (voice, video, and desktop sharing), and personal phone call control management, such as call forwarding, call hold, etc. The user can manage all of these services via a convenient web browser interface or desktop client. We evaluated the OpenScape UC Application server for its security countermeasures without the use of additional security gateways or firewalls in the deployed topology.

We tested the UC Application Server to ensure that during any attack, changes that needed to be pushed could be completed as required. This can be extremely vital if, for example, another system is compromised, and an alert or background change can be pushed to all phones notifying the entire office and branches of the issue silently. If an attack is underway, UC Application Server needs to remain functional.

This report provides results that were used to qualify UC Application Server as Certified Secure. OpenScape Deployment Service V7 achieved the Miercom Certified Secure rating that is reserved for products that score in the top 30% of a product class and can prove their resiliency to compromising security threats.

# 3.0 Use of the Data in this Report

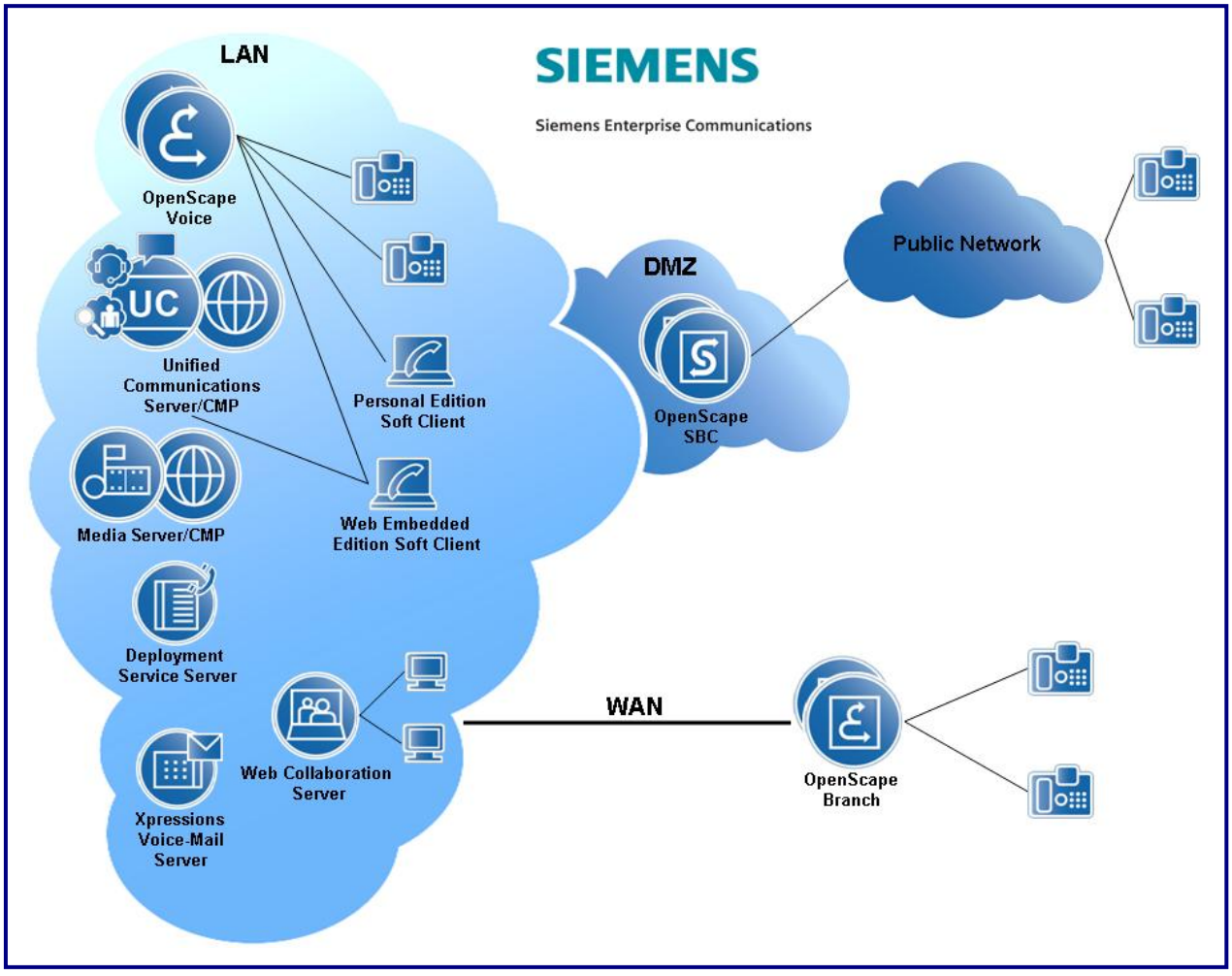
All tests conducted on OpenScape products were inside the internal network. No external security measures were used. We stressed the inherent resiliency of the hardened OpenScape systems themselves. Miercom does not offer a warranty of fitness or suitability for this product in a customer's environment based on this testing alone, without Miercom additionally conducting the site assessment and integration for the deployment.

# 4.0 OpenScape UC Application Server Deployment Diagram

The following representative diagram depicts an enterprise network for OpenScape Voice with OpenScape UC Application. The OpenScape Voice system was connected in a “High Availability” configuration that would be typical in a larger enterprise configuration. The High Availability configuration consists of contingency measures to maintain availability if the OpenScape system application fails. This includes a backup Media Server, which is a sub-server in the OpenScape system. Security assessment of OpenScape UC Application Server was made without use of any security gateways, firewalls or SBC in the deployed topology.

Exact configuration and tools used in testing for this project is proprietary for the protection of the security testing program and the vendor being tested.

## Logical Configuration of OpenScape Unified Communications Application Server V7



# 5.0 How We Did It

To conduct the following tests, we used a combination of test tools including customized proprietary test scripts, commercial and open-source vulnerability scanning tools and security assessment products. We used a VoIP network infrastructure typical of a mid-sized enterprise to support OpenScape Voice as the primary server and UC Application Server as the system under test as much as possible. We evaluated the viability of each attack and the risk of compromise of OpenScape products resulting in recommending compensating measures to rectify or mitigate the suspect vulnerabilities. Additional details are shown in the OpenScape Deployment Diagram.

The objective was to compromise the ability of the OpenScape UC Application to successfully deliver real-time voice communications and to gain surreptitious access to the OpenScape infrastructure for the purposes of exploiting the system. The OpenScape system was evaluated inside the internal network. The tests were conducted without external security countermeasures employed to the OpenScape systems. The objective was to assess the resiliency of the management platform first and then subsequently assess the security of the interaction of the UC Application Server and the underlying network.

OpenScape UC Application Server was evaluated while it was installed in a standard quality assurance test environment, meaning all the recommended deployments and guidelines, as per Siemens documentation was followed with monitors for every metric tested under constant scrutiny. We did a perimeter assessment to identify paths through the network that could be used to assault the systems by a potential attacker. We identified paths that merited exercising to find flaws. We then exercised these paths as potential attack vectors that are further detailed within this report.

A series of Mu Dynamics, Metasploit, and Miercom proprietary tests were performed to analyze the security of the Deployment Service. The testing subjected the OpenScape UC Application Server to a methodical series of vulnerability analyses and scans. Tests included a complex set of exploits distributed by security tools and scripts to challenge the capabilities of the product. Without compromising its integrity, the OpenScape UC Application Server blocked all attempted attacks and proved to
be resilient.

Penetration test tools used to run attacks/exploits, security scans including protocol interaction with mutated traffic, common vulnerability exploit tests, Denial of Service (DoS) and protocol mutation attacks included proprietary test scripts and the open-source security assessment products, Offensive Security, and Mu Dynamic's Mu Studio Security. The Mu analyzer provides a complete service assurance solution for determining the reliability, availability and security of IP-based applications and services.

The Mu methodically probes for vulnerabilities using attack vectors. The Mu analyzer tested the resiliency of the UC Application Server from protocol mutations, published vulnerabilities and external attacks using test cases and custom scripts. The Mu solution is highly automated with lights-out fault isolation. It can help speed the remediation of software flaws by providing actionable reports and complete data on any faults.

We evaluated the viability of each attack and the risk of compromise to the SUT and recommended compensating measures to rectify or mitigate the suspected vulnerabilities.

# 6.0 Attacks against OpenScape UC Application

## 6.1 DHCP INFORM Mutation Attack

### Description

DHCP is a protocol to allow a computer to automatically receive an IP address. The proper transfer of the IP address data is of paramount importance in keeping a device performing smoothly. Corruption of this data could cause a failure of the device.

### Configuration

The test was configured to attack the OpenScape UC Application Server interface using DHCP INFORM messages.

### Purpose

To determine whether the dynamic IP address could be corrupted by the attack from the mutated DHCP protocol.

### Observations

The DHCP INFORM attack directed a total of 11,843 protocol mutations generated from 393 variants contained in 1 test suite. During the attack we did not see any negative effect on the performance or security of the server. The UC Application Web GUI was fully navigable, performing its normal operation. New calls were able to be placed from within the application to users in the network. When an incorrect number was dialed, the media server correctly notified the caller with the incorrect number announcement.

### Analysis

**PASS**

All attack vectors were handled successfully and no faults were reported. OpenScape UC Application Server dropped all mutated packets and did not send any error messages. No vulnerabilities in the DHCP protocol implementation at the UC Application Server were detected. All server functionality and features remained fully operational during and after the attack.

## 6.2 IPv4 Mutation Attack

### Description

IPv4 is the most widely used Internet communications protocol for the proper formatting of digital messages and procedures for the exchange of messages between computing systems. It is a connectionless protocol for use with the Ethernet family of networking technologies for local area networks. It functions on the “best effort delivery” philosophy, in that the network does not guarantee the delivery of the data or a quality of service level. The users of this protocol are not guaranteed a specific bit rate or delivery time. It is dependent on the current traffic load. The IPv4 protocol is limited in its IP addresses due to its 32-bit architecture. These IP addresses are vulnerable to attack and corruption. If this data were corrupted, it could lead to device failure. We tested to determine whether the IP address could be corrupted by the attack from the mutated IPv4 protocol.

### Configuration

The test is configured to attack the UC Application Server interface using IPv4 normal and fragmented datagrams.

### Purpose

The purpose of the IPv4 attack is to determine whether the UC Application Server will appropriately block all malformed packets.

### Observations

The IPv4 attack directed a total of 31,129 protocol mutations generated from 113 variants contained in 1 test suite. During the attack we did not see any negative effect on the performance or security of the server. The Unified Communications Web GUI was fully navigable, performing its normal operation. New calls were able to be placed from within the application to users in the network. When an incorrect number was dialed, the media server correctly notified the caller with the incorrect number announcement.

### Analysis

**PASS**

All attack vectors were handled successfully and no faults were reported. The UC Application Server dropped all mutated packets and did not send any error messages. No vulnerabilities in the IPv4 protocol implementation at the UC Application Server were detected. All server functionality and features remained fully operational during and after the attack.

## 6.3 ICMPv4 Mutation Attack

### Description

Internet Control Message Protocol (ICMP) is a communications protocol for the proper format of digital messages and procedures for the exchange of messages between computing systems. Its main purpose is to work within the computer operating system of the network, monitoring the status of a requested service, host or router, and determining their availability. Another use of this protocol is to relay messages. It can also be used as a diagnostic ping or network tracer tool. IP addressing is crucial to its correct operation. Any corruption of the IP address could cause failure of the system.

### Configuration

The test is configured to attack the UC Application interface using ICMPv4 echo requests and fragmented echo requests.

### Purpose

The purpose of the ICMPv4 attacks is to determine whether the OpenScape UC Application Server will block the mutated packets and maintain full system functionality.

### Observations

The ICMPv4 attack directed a total of 42,981 protocol mutations generated from 298 variants in 2 test suites. During the attack we did not see any negative effect on the performance or security of the server. The UC Application Web GUI was fully navigable, performing its normal operation. New calls were able to be placed from within the application to users in the network. When an incorrect number was dialed, the media server correctly notified the caller with the incorrect number announcement.

### Analysis

**PASS**

All attack vectors were handled successfully and no faults were reported. The UC server dropped all mutated packets and did not send any error messages. No vulnerabilities in the ICMPv4 protocol implementation at the UC Application Server were detected. All server functionality and features remained fully operational during and after the attack.

## 6.4 UDP Mutation Attack

### Description

User Datagram Protocol (UDP) is an Internet Protocol where computer applications can send messages, hereafter referred to as datagrams, to other computers or devices on the network, without prior communications for setting up a transmission channel or data path. As a result of this process, datagrams could arrive out of order or go missing. To achieve host to host communication, UDP uses datagram sockets which combine an IP address and port address. A successful attack on the IP or port address could lead to system failure and lost data.

### Configuration

The UDP protocol mutation attack was launched against the UC Application Server interface on port 9091 which was identified earlier during scans.

### Purpose

We would determine whether the UC Application Server's use of the UDP protocol could be corrupted by the mutated attack.

### Observations

The UDP attack directed a total of 6,411 protocol mutations generated from 106 variants in 1 test suite. During the attack we did not see any negative effect on the performance or security of the server. The OpenScape UC Application Web GUI was fully navigable, performing its normal operation. New calls were able to be placed from within the application to users in the network. When an incorrect number was dialed, the media server correctly notified the caller with the incorrect number announcement.

### Analysis

**PASS**

All attack vectors were handled successfully and no faults were reported. The UC Application Server dropped all mutated packets and did not send any error messages. No vulnerabilities in the UDP protocol implementation at the UC Application Server were detected. All server functionality and features remained fully operational during and after the attack.

## 6.5 SSL/TLSv1 Mutation Attack

### Description

This suite contains test cases for Transport Layer Security, version 1 (TLSv1) messages. Like SSL, TLS ensures privacy for Internet communications between applications and users and prevents third party eavesdropping and/or content tampering of any message. TLSv1 operates between the Transmission Control Protocol (TCP) and an application, such as HTTP.

A TLSv1 message contains a Record layer, which communicates with the higher-layer application and structures the message, and a component, which is a separate protocol that performs a specific task or carries a specific type of data.

### Configuration

The SSL/TLS protocol mutation attack was launched against the OpenScape UC Application interface.

### Purpose

We would determine whether the UC Application's use of SSL could be corrupted by the mutated attack.

### Observations

The UC Application Server firewall completely prevented the SSL protocol attacks from running. We were able to direct attacks using TLSv1 to port 8443 identified earlier in scans. The attack sent 2,176 test cases with 95,498 variations in 1 test suite. During the attack we did not see any negative effect on the performance or security of the server. The UC Application Web GUI was fully navigable, performing its normal operation. New calls were able to be placed from within the application to users in the network. When an incorrect number was dialed, the media server correctly notified the caller with the incorrect number announcement.

### Analysis

**PASS**

All attack vectors were handled successfully and no faults were reported. The UC Application server completely blocked SSL attacks, dropped all mutated TLSv1 packets and did not send any error messages. No vulnerabilities in the TLSv1 protocol implementation at the UC Application server were detected. All server functionality and features remained fully operational during and after the attack.

## 6.6 Nmap

### Description

Nmap was used to scan each system in the UC Application environment for open ports. A single IP address or an IP address range may be inputted into Nmap to reveal vulnerable devices on the network. A system with open ports can pose a threat if they are not securely implemented.

### Configuration

Nmap was configured to do a full system scan on the UC Application Server.

### Purpose

The purpose of Nmap scanning is to identify all vulnerable ports, and all services running with advertisement behind those ports. By running an in-depth discovery scan, we can identify any and all vectors and surfaces susceptible to penetration, exploitation, or general attack.

### Observations

We discovered services running behind 22, 443, 8443, and 9091.

### Analysis

**PASS**

All ports except 9091 indicate SSL transmissions or TLS transmissions, which are highly secure. TLS transmissions if enforced are nearly unbreakable in a reasonable time period, and generally considered not worthwhile to even bother deciphering. The system uses 128-bit keys, which could take an average single core processor millions of years to guess. Because of this general exploits will be attempted but are likely to prove largely unsuccessful, and the service behind port 9091 could not be accurately fingerprinted. Positive results, if any, will be discovered through attack and not discovery.

## 6.7 Metasploit

### Description

Offensive Security is a Linux-based penetration organization catered towards security and vulnerability assessment. Metasploit has the abilities to gather and map network information, identify vulnerabilities, analyze web applications, perform digital forensics and reverse engineering, and attempt automated stealth scans and penetration.

### Configuration

Several information gathering, vulnerability identification and penetration tools were used against the UC Application Server to find all potential attackable surfaces.

### Purpose

To identify, and attempt all commonly known exploits to ensure the system is not outdated.

### Observations

We were unable to gather any information or detect any vulnerabilities that would help lead to a successful penetration of the server. The primary filtered ports (requiring proper credentials) were 443 and 8443. Penetration attempts were made on these ports to determine if a common exploit was not properly addressed. In our testing, no penetration attempts succeeded.

### Analysis

**PASS**

No internal access was gained and the portal stayed online and fully functional throughout the attack. If Metasploit is successful, it typically can gain root or shell access to a server. Since Metasploit is a common framework, it is built on release information by the community as well as by the manufacturers. In order to stay “Metasploit-proof,” all software must be kept up to date, as the framework itself updates hourly. In total 138 penetration attempts were made against UC Application Server.

## 6.8 Web Application, Audit, and Attack

### Description

*Web Application, Audit, and Attack* is a framework to find and exploit web application vulnerabilities that is easy to use, extend, and tailor to specific scenarios. Tools include discovery, evasion, audit, grep, mangle, and output. This framework is designed specifically to perform a very intensive HTTP audit.

### Configuration

Several information gathering, vulnerability identification and penetration tools were used against the UC Application Server to find all potential attackable surfaces.

### Purpose

The W3AF is capable of committing layer 5, 6 and 7 discoveries and attacks. This is vital because on high level complex software this is the likeliest place of vulnerability. This will also identify the most fuzzing and injection vulnerabilities.

### Observations

We observed several no faults.

### Analysis

**PASS**

Port 443 appears to be secure and not vulnerable to standard dictionary-based fuzzing, or injections.

## 6.9 Miercom Test Suite

### Description

The Miercom test suite contains a plethora of proprietary protocols and test methodologies developed over the years to audit and commit attacks on various kinds of systems. Not only do we test standard server capabilities, but we can also look at high layer applications and services in an attempt to break or otherwise disrupt service.

### Configuration

For this specific system all of the suites maintained similar configurations. Attacking right from the front of the server on its normally accessible port 443, all known potential vulnerabilities and exploits were attempted. This includes various dictionary attacks, SQL injections, web fuzzing, web application firewall penetration, and many others.

### Purpose

The purpose of these tests is to disrupt communications or accessibility in any reasonable way; however the ultimate goal is to obtain internal privileged access.

### Observations

None of the tests proved successful. The OpenScape UC Application Server subverted all attacks or stopped any attempt to penetrate the system.

Note: The product tested is capable of implementing third party security certificates. However, we tested with certificates derived internally as we did not have Internet access in the test bed to access/verify third party security certificates.

### Analysis

**PASS**

Both manual and automated web spidering was attempted. All manual fuzzing, injection, and web application firewall penetration tests proved unsuccessful.

# 7.0 Other Analyses Conducted

The following is a summary of the compound exploits and other analysis conducted in the OpenScape UC Application Security Assessment:

**Port Scanning and Enumeration** – Open source tool Nmap was used to scan the OpenScape servers for unused open ports, OS fingerprinting, version numbers, supported protocols, running services, and other information that could be used to attack the OpenScape servers.

**Integer and Buffer Overflow Tests** – These mutations add, insert or replace input with a large number of random bytes, in an effort to cause data to exceed the boundaries of its specified location. Overflow attacks exploit computer methods used to store integers, which are variables that represent real, non-fractional numbers. We represent integers in decimal format (using 10 numerals, 1-10), but computers store integers in binary format (using two numerals, 1 and 0). If the operation produces a value larger than the maximum integer size for the data, an integer overflow occurs. Miercom verified that these potential integer overflow conditions did not cause buffer overflows.

**Fragmented Attacks** – Fragmented packets were used to infiltrate and cause degradation in server performance. Such fragmented packets can get past Access Lists (ACLs) in stateless packet filtering deployment and be further used to cause DoS. Several types of attacks - teardrop, overlapping fragment and tiny fragment - were tested with the OpenScape system. No vulnerabilities were found after the fragmented attacks were applied.

### Analysis

**PASS**

No abnormalities were detected in these directed attacks.

# 8.0 Denial of Service Attacks

Denial of Service (DoS) attacks were generated and directed at the OpenScape UC Application server to gain insights into reliability, availability and security of service in the face of DoS attacks or extreme amounts of service level traffic. While attacking the OpenScape servers, our objective was to saturate them to the extent that they could not respond to legitimate traffic, so that they would become unresponsive and slow, or that they would crash or reboot, which all can lead to failures at the SIP phones.

Metasploit, Offensive Security and Mu Studio Security were used to configure 28 different DoS attacks with fixed and randomized source ports (IP and MAC addresses). TTLs, TCP sequence numbers, payload, user-defined TCP header values, randomized protocol types and other values for the attack packets were also configured. Attack patterns included different start/end rates (packets /sec), duration of attacks and number of attack iterations. Target availability and response time was verified at defined intervals during the attacks using ICMP.

The OpenScape UC Application Server was preconfigured and hardened to counter DoS attacks.

A partial list of DoS attacks used and their results are discussed in the following sections:

## 8.1 ICMP Flood DoS

### Description

These IP packets comprise an ICMP flood, which is a DoS attack that is also known as a ping flood or Smurf attack. During the attack, the Mu analyzer sent large amounts of ICMP packets to the target system in an attempt to crash its TCP/IP stack and cause it to stop responding to TCP/IP requests.

### Configuration

The DoS attack consisted of 100,000 packets per second and was directed at the OpenScape UC Application Server interface. The ability of the UC Application Server to maintain its functionality was monitored.

### Purpose

The purpose of the test is to verify the firewall effectiveness of the UC Application Server when faced with large amounts of ICMP packets.

### Expected Results

It is expected that the OpenScape UC Application will detect and block the ping flood with no disruption in maintaining services.

### Observations

All DoS attack attempts were dropped by the UC Application Server V7, and attempts to compromise its operation were unsuccessful.

### Analysis

**PASS**

The UC Application Server functionality was maintained throughout the test. The web GUI was fully accessible without any disruptions.

## 8.2 IPv4 DoS Attacks

### Description

The attack consisted of IP packets containing datagrams and fragmented datagrams, which is a DoS attack designed to flood the system. During the attack, the Mu Studio Security sent large amounts of IPv4 packets to the target system in an attempt to crash the TCP/IP stack and cause it to stop responding to TCP/IP requests.

### Configuration

The DoS attack consisted of 100,000 packets per second and was directed at the OpenScape UC Application Server interface. The ability of the UC Application Server to maintain its functionality was monitored.

### Purpose

The purpose of the test is to verify the firewall effectiveness of the UC Application Server when faced with large amounts of packets.

### Expected Results

It is expected that the UC Application Server will detect and block the TCP datagram flood with no disruption in maintaining services.

### Observations

All DoS attack attempts were dropped by the UC Application Server V7, and attempts to compromise its operation were unsuccessful.

### Analysis

**PASS**

The UC Application Server functionality was maintained throughout the test. The web GUI was fully accessible without any disruptions.

## 8.3 UDP DoS Attack

### Description

The attack consisted of UDP packets containing datagrams and fragmented datagrams, which is a DoS attack designed to flood the system. During the attack, the Mu analyzer sent large amounts of UDP packets to the target system in an attempt to crash the TCP/IP stack and cause it to stop responding to TCP/IP requests.

### Configuration

The DoS attack consisted of 100,000 packets per second and was directed at the OpenScape UC Application Server.

### Purpose

The purpose of the test is to verify the firewall effectiveness of the UC Application Server when faced with large amounts of packets.

### Expected Results

It is expected that the UC Application Server will detect and block the UDP datagram flood with no disruption in maintaining services.

### Observations

All DoS attack attempts were dropped by the UC Application Server V7, and attempts to compromise its operation were unsuccessful.

### Analysis

**PASS**

The UC Application Server functionality was maintained throughout the test. The web GUI was fully accessible without any disruptions.

# 9.0 Viruses, Worms and Botnets

## Description

Viruses, worms and botnets are specific variations of DoS attacks. The objective is to identify weaknesses that could affect reliability, availability and security of the network. A botnet attack would be characterized by multiple systems attempting to access the OpenScape UC Application Server, but with much higher than normal frequency, causing the system to become unavailable. Blacklist library size can play a key role in mitigating this type of attack. The Witty worm exploits a firewall application vulnerability, targeting systems that have employed proactive countermeasures, and includes a destructive payload. To propagate the worm, infected hosts send packets as fast as an Internet connection will allow. The Slammer worm also exploits similar buffer overflow vulnerabilities, infecting hosts and randomizing destination IP addresses to enable further propagation.

## Test

The Mu Studio Security analyzer was used to evaluate the effect of these attacks on the server's various interfaces. Interface availability was monitored, as well as any undesirable effect on system operation. VoIP calls were placed throughout the test at a rate of one call per second, and the call completion rate was also monitored. By randomizing the source port and address during these attacks, the behavior of a botnet was emulated. Witty and Slammer worm DoS attacks were configured with random source and destination IPs.

## Observations

**PASS**

These attacks were unsuccessful in disrupting the operation of the system. Real-time monitoring and post-test analysis of the server logs indicated successful blocking of the attacking source addresses. The OpenScape servers and interfaces remained available, and VoIP calls continued to be placed successfully during the attacks.