Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

Note

# VoIP Lab as a Research Tool in the EUX2010sec Project

Note no DART/08/10

Author Lars Strand

Date 28. April 2010

## The author

Lars Strand is a research fellow working on a PhD degree at the Norwegian Computing Center. His research area is network security in VoIP, with a focus on the SIP protocol.

## Norwegian Computing Center

Norsk Regnesentral (Norwegian Computing Center, NR) is a private, independent, non-profit foundation established in 1952. NR carries out contract research and development projects in the areas of information and communication technology and applied statistical modeling. The clients are a broad range of industrial, commercial and public service organizations in the national as well as the international market. Our scientific and technical capabilities are further developed in co-operation with The Research Council of Norway and key customers. The results of our projects may take the form of reports, software, prototypes, and short courses. A proof of the confidence and appreciation our clients have for us is given by the fact that most of our new contracts are signed with previous customers.

**Norsk Regnesentral**
Norwegian Computing Center
Postboks 114, Blindern
NO-0314 Oslo, Norway

**Besøksadresse**
*Office address*
Gaustadalléen 23
NO-0373 Oslo, Norway

**Telefon** · telephone
(+47) 22 85 25 00
**Telefaks** · telefax
(+47) 22 69 76 60

**Internett** · internet
www.nr.no
**E-post** · e-mail
nr@nr.no

| | |
|---|---|
| **Title** | **VoIP Lab as a Research Tool in the EUX2010sec Project** |
| **Author** | **Lars Strand** <lars.strand@nr.no> |
| Quality assurance | Arne-Kristian Groven, Wolfgang Leister |
| Date | 28. April 2010 |
| Publication number | DART/08/10 |

## Abstract

As part of the research project EUX2010sec at NR, a Voice over IP testbed has been set up. The testbed has a central role in the research project that enables us to do testing, experimentation and measurements of VoIP protocols and VoIP scenarios. This white paper describes the equipment available, both server hardware and UAs (phones), software components and various network configuration setup of the testbed.

| | |
|---|---|
| Keywords | VoIP, lab, SIP, network |
| Target group | Users of the EUX2010sec lab and other interested in VoIP |
| Availability | Open |
| Project | EUX2010sec |
| Project number | 180054/S10 |
| Research field | VoIP, security |
| Number of pages | 14 |
| © Copyright | Norwegian Computing Center |

# Contents

# 1 Introduction

The overall goal of the EUX2010sec[1] project is to improve both the security level and the security awareness when developing, installing, and using open source solutions for VoIP/PBX and multimedia. To reach this goal a testbed has proven necessary as a research approach, as outlined by Fritsch et al. (2009). This testbed has been set up at NR, in close cooperation with our project partners.

# 2 Purpose

The EUX2010sec testbed enables us to have a VoIP infrastructure for experimentation, analysis and testing of VoIP components in various scenarios. This gives us an advantage over a pure theoretical of simulation approaches, since the performance of a VoIP installation has many deciding factors, like network utilization and congestion, the network architecture, protocols and security mechanisms.

We have defined five goals for the testbed:

1. Use the testbed as a training arena for researchers, project partners and third party vendors.
2. Replicate VoIP installations based on project partner's configurations, requirements and technologies. Test these installations against the project partner's security policy. If vulnerabilities should be found, we implement VoIP-related security mechanism, and raise awareness for new re-engineering steps.
3. Use the testbed to try out VoIP-related attacks in a controlled environment.
4. Implement a configuration management that enables the reuse of a given testbed configuration to researchers, and third party vendors.
5. Develop best practises, and more secure VoIP configurations based on the experiences we have gained in the preceding steps.

# 3 Equipment

The testbed consists of network equipment, server hardware, and a variety of different hardphone models (UAs). The testbed is depicted in Figure 1 on the following page.

## 3.1 Hardware

The available servers span from high-end servers capable of serving several thousand SIP requests per seconds, to ordinary tower PCs that function as administrative and/or

[1] Project homepage: `http://eux2010sec.nr.no/`

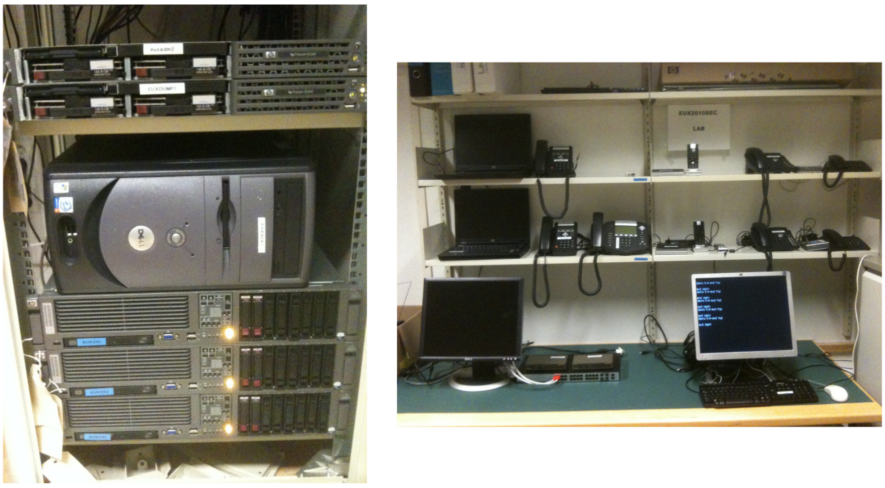

Figure 1. The testbed in the lab. The servers in the rack to the left, and the phones to the left.

attack nodes. A full list of the server hardware can be found in Table 1 on the next page.

All of our HP server have *iLO*[2] for remote management. Since all iLO interfaces are configured to use the internal AD-network segment, they can only be accessible from NRs internal network (or by using SSH port-forwarding).

## 3.2 Phones

The testbed has several phones, aka. User Agents (UA), available. Several different softphones can be tested, and we have 6 different hardphones models from 3 different vendors. The total number of hardphones counts 12, since we have two sets of each model. For a detailed list, see Table 2 on page 8.

## 3.3 Network

We have several options when connecting the testbed to the Internet. Currently we have two 1Gbps network connections to the Internet, but more can be added if needed. Internally, the testbed uses RFC1918 (Rekhter et al., 1996) IPv4 addresses. The testbed's gateway uses network address translation (NAT) to route traffic to/from the Internet.

We can easily add support for VPN, if needed. The same goes for POTS connectivity (ordinary "old" telephony support), since we have several PSTN ports available in the lab.

For a complete list of network equipment, please consult Table 3 on page 9.

[2] HP Integrated Lights-Out (iLO): `http://h18013.www1.hp.com/products/servers/management/remotemgmt.html`

| Hostname | Role/description | Hardware | Image |
|---|---|---|---|
| eux1.eux.lab,<br>eux2.eux.lab | Attack, test and measure probe | Custom built 2.4GHz, 768MB RAM, 80GB disk. | |
| euxadm1.eux.lab | Administrative services like DNS, LDAP, configuration management (Subversion), monitoring, NTP, TFTP and SMTP | Dell Dimension 2400, 2.8GHz, 1GB RAM, 40GB disk, 1x network interface | |
| euxadm2.eux.lab | Virtual Machine Server hosting virtual servers | HP DL360 G4, 2x Xeon 3.0GHz, 1GB RAM, 2x75 GB in RAID 0, 2x network interface | |
| euxcs1.eux.lab | Connection Server (CS) / Session Border Controller (SBC) | HP DL380 G5, 2x Xeon Dual-Core 2.0GHz, 6GB RAM, 2x 2.5" SAS 73.4GB in RAID 1, 4x network interface | |
| euxdump1.eux.lab | Performs network traffic dump | HP DL360 G4, 2x Xeon 3.0GHz, 4GB RAM, 2x75 GB RAID 0, 2x network interface | |
| euxss1.eux.lab,<br>euxss2.eux.lab | Session Service Servers | HP DL380 G5, 2x Xeon Dual-Core 2.0GHz, 6GB RAM, 2x 2.5" SAS 73.4GB in RAID 1, 4x network interface | |
| switchb1.eux.lab,<br>switchb2.eux.lab | For testing various PBX switchboard | Compaq 6715b Laptop | |

Table 1. Server hardware available in the testbed.

| Model | Description | Image |
|---|---|---|
| 2x Linksys WIP330 | WLAN capable SIP phone (WEP only). | |
| 2x Polycom Sound-point IP330 | SIP capable phone, PoE. | |
| 2x Polycom Sound-point IP550 | SIP capable phone, PoE. | |
| 2x SNOM M3 | DECT wireless SIP phone (with gateway). | |
| 2x SNOM 300 VoIP Phone | SIP capabale phone, PoE. | |
| 2x DORO Congress 100 | POTS phone (non-VoIP). Used in conjunction with the Linksys SPA2102 VoIP gateway. | |
| 2x Linksys SPA2102 | POTS phone adapter with VoIP (SIP) gateway. | |

Table 2. VoIP hardphones available in the testbed.

| Model | Description | Image |
|---|---|---|
| 1x DLink DGS1005D | 5 port 10/100BASE-TX network switch | |
| 2x Linksys WRT54GL | Wireless router (802.11g) with 4 port 10/100BASE-TX network switch and 1 port 10/100BASE-TX WAN | |
| 2x DLink DES1008P | 8 port 10/100BASE-TX network switch with 4 PoE ports | |
| 2x DLink DES1228P | 24 port 10/100BASE-TX network switch with PoE and 2 port 10/100/1000T | |
| 1x Linksys SRW2008 | 8 port 10/100/1000T network switch (managed) | |

Table 3. Network equipment available in the testbed.

# 4 Software

There are several software components installed and in use in our testbed. We use primarily open source software, since the open source development model is an important aspect of our research project. For specific tests we can also use proprietary software.

## 4.1 VoIP

For VoIP software server components, we use primarily the Asterisk PBX[3]. Asterisk is the most popular and industry adopted open source telephony server (PBX). We have also tried and used other open source PBXs, like Kamilio[4] and SIP Express Router (SER)[5].

We have also tested and used several VoIP clients (softphones), such as X-Lite[6], and several open source clients like Ekiga[7], Twinkle[8], kphone[9] and SIP Communicator[10].

## 4.2 Network

The ability to dump network traffic for off-line examination and analysis is important. To do this, we use a feature called "port-mirror" in the testbed's access switch, as depicted in Figure 2 on the following page. This feature duplicates all traffic from one Ethernet port to another. Another server euxdump1 is passively listning to and saving all network

[3] Asterisk PBX homepage: http://www.asterisk.org
[4] Kamilio (former OpenSER) homepage: http://www.kamailio.org/
[5] SIP Express Router (SER) homepage: http://www.iptel.org/ser/
[6] X-Lite from CounterPath: http://www.counterpath.com/x-lite.html
[7] Ekiga homepage: http://ekiga.org/
[8] http://www.xs4all.nl/~mfnboer/twinkle/
[9] KPhone homepage: http://sourceforge.net/projects/kphone/
[10] http://sip-communicator.org/

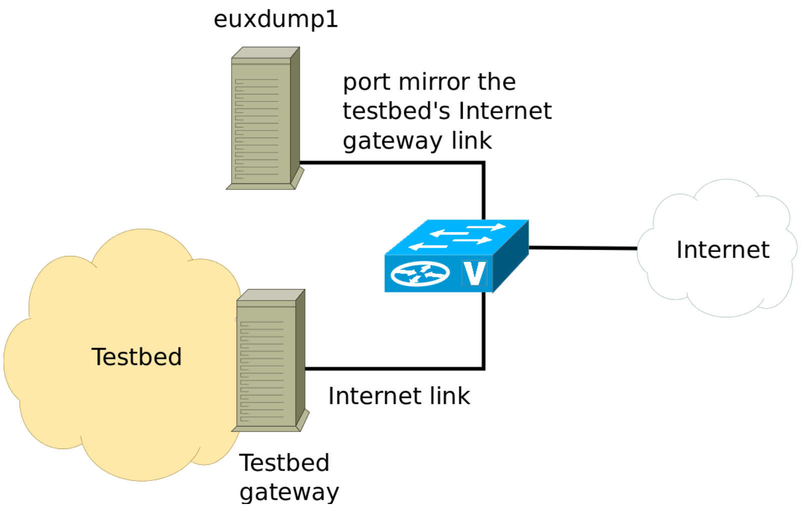

Figure 2. We use “port-mirror” to duplicate all network traffic to/from the testbeds Internet gateway to enable passive network dumping of all traffic.

traffic to file from the port-mirrored Ethernet link. In practice, any network interface in the testbed can be port-mirrored, but we currently have set it up to port-mirror all traffic from the testbed’s Internet gateway.

Running the network dump on the same server as the test are executed may effect the results (e.g., for performance testing). Therefore, the dump server `euxdump1` runs the program “tcpdump”[11] on the listening passive interface, and saves all network traffic to a file for off-line examination and analysis later on.

## 4.3 Administrative services

We have several administrative services running in the testbed that can be found in a normal real-life industrial VoIP network. These services include DNS, LDAP, SMTP (mail), monitoring tools and configuration management (Subversion). These administrative services are hosted on the servers `euxadm1` and `euxadm2`.

### 4.3.1 DNS

The testbed has internal DNS servers. Both as resolving, caching name server for the internal hosts in the testbed, but also serving the internal “eux.lab” domain (both forward and reverse zones) for the RFC1918 IPv4 addresses used. In the “Scenario 2” setup, as described in Section 6.2, we also configured sub-domains (“*.companyA.eux.lab”, “*.companyB.eux.lab”) for the different sub-networks.

We use the open source software BIND[12], the most widely used DNS software on the Internet. The server `euxadm1` is set up to be master and `euxadm2` as slave. All relevant configuration for DNS can be found under `/etc/bind/` on both servers.

[11] TCPdump/libpcap homepage: `http://www.tcpdump.org/`

[12] BIND homepage: `http://www.isc.org/software/bind`

### 4.3.2 LDAP

The "Lightweight Directory Access Protocol", or LDAP, is used to query and modify data using directory services. A directory can contain objects that have a set of attributes that are stored in a hierarchical manner. The most common industrial use for LDAP is user authentication and "address-book lookup" in email clients.

The LDAP in the testbed functions as user authentication lookup. All project members, partners and training students that need access, will have their user information stored in LDAP. This simplifies administration of users in the testbed, and enables us to test lookup mechanism in the phones that support that (like the Polycom models and softphones).

### 4.3.3 SMTP

Many common alerts are sent using SMTP (email). This also includes system services running locally on each server. To prevent each server to store its own email locally, we have SMTP server running Postfix. The internal DNS have a MX-record that points to the administration servers.

The SMTP support in our testbed enables us to replicate services that require email. This includes monitoring alerts and voicemail sent over SMTP.

### 4.3.4 Monitoring

We have a range of common monitoring tools configured. We use Munin[13] to analyze resource trends when performing tests. We use Nagios[14] to test alert monitoring. We also use MRTG[15] to monitor the network utilization for the network equipment that supports SNMP.

These monitoring tools (and others) enables us to test the monitoring capabilities of the industry setup and to test if these fulfills the security requirements given.

### 4.3.5 Configuration management

We use Subversion[16] for all configuration management. All relevant configurations in the testbed are stored in Subversion. This quickly enables us replicate any given setup. Each scenario has its own "branch" in Subversion, so that we easily can distinguish between them. Subversion gives us control over the configuration used in the testbed.

# 5 Public phone service

We have 22 Norwegian public telephone numbers assigned to the lab. Of these, 20 are are ordinary geographical (address) linked land-line phone numbers, and two are non-geographical (starting with 852). All phone numbers are assigned to Ventelo on behalf of

[13] Munin homepage: `http://munin-monitoring.org/`

[14] Nagios homepage: `http://www.nagios.org/`

[15] The Multi Router Traffic Grapher (MRTG) homepage: `http://oss.oetiker.ch/mrtg/`

[16] Subversion homepage: `http://subversion.tigris.org/`

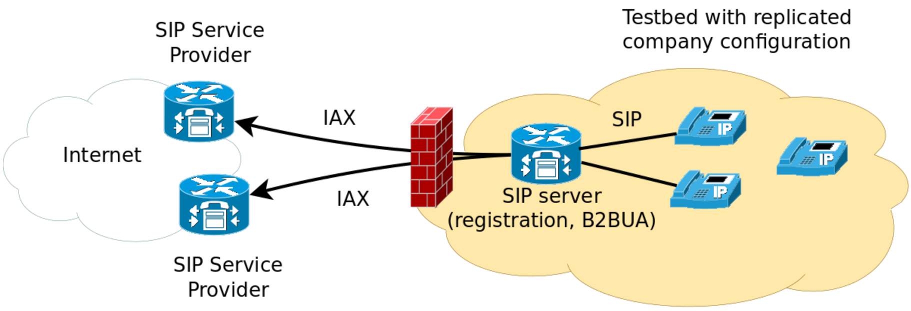
SIP Service Provider
Testbed with replicated company configuration
Internet
IAX
SIP
IAX
SIP server (registration, B2BUA)
SIP Service Provider
IP

Figure 3. Testbed scenarios 1, a single server functioning as a SIP registrar and back-to-back user agent (B2BUA).

one of our project partners.

# 6 Testbed scenarios

Currently, we have tested two VoIP scenarios. Both scenarios are provided by project partners. The scenarios are replica of real-world VoIP installations, with the same configuration and software components as their real-world counterparts. The setup of each scenario was done in cooperation with the relevant project partner, to ensure the correctness of the setup.

## 6.1 Scenario 1

The first testbed scenario was a VoIP replica of a VoIP installation from one of our partners, as depicted in Figure 3. We installed the same network configuration, the same operating system, the same VoIP software and configuration. The test results we acquired from this scenario were used as data for the work by Hagalisletto and Strand (2008).

## 6.2 Scenario 2

The second scenario was a replica of the a configuration used by our project partner for Buskerud Fylkeskommune (Buskerud county), as depicted in Figure 4 on the following page. We have used this scenario for internal training, as part of a VoIP course exercise. Further tests have resulted in work by Hagalisletto et al. (2009), and by Hagalisletto and Strand (2010).

# 7 Conclusion

The testbed has proven important to the EUX2010sec project as a research tool and approach. It is also an important training tool, to better understand the complex interworkings of the different VoIP components and protocols. The testbed is also a place where the project partners can experiment with different configuration before deploying new

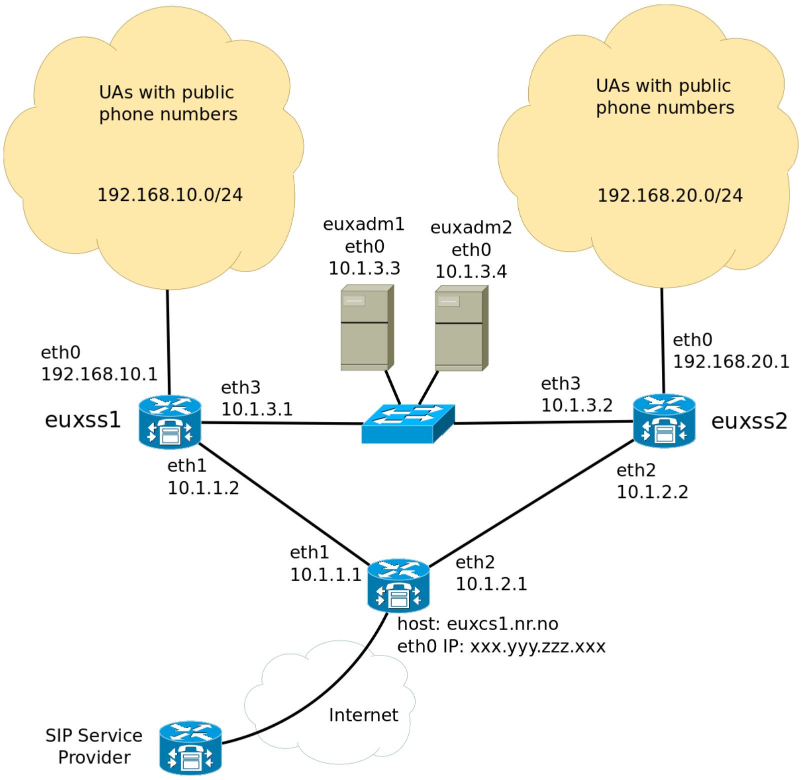

Figure 4. Testbed scenarios 2 with two departments and a failover backbone in between them. All servers running Asterisk with SIP and RTP and public phone numbers.

solutions in their own network.

# Acknowledgments

This research is funded by the EUX2010sec project in the VERDIKT framework of the Norwegian Research Council (Norges Forskningsrad, project 180054). The author would like to thank Wolfgang Leister for comments on the earlier drafts of this paper.

# References

Fritsch, L., Groven, A.-K., Strand, L., Leister, W., and Hagalisletto, A. M. (2009). A Holistic Approach to Open Source VoIP Security: Results from the EUX2010SEC Project. *International Journal on Advances in Security*, (2&3):129–141. 5

Hagalisletto, A. M. and Strand, L. (2008). Formal modeling of authentication in SIP registration. In *Second International Conference on Emerging Security Information, Systems and Technologies SECURWARE '08*, pages 16–21. IEEE Computer Society. 12

Hagalisletto, A. M. and Strand, L. (2010). Designing attacks on sip call set-up. *International Journal of Applied Cryptography*, 2(1):13–22(10). Available from: http://

inderscience.metapress.com/link.asp?id=jh437k6747064307. 12

Hagalisletto, A. M., Strand, L., Leister, W., and Groven, A.-K. (2009). Analysing protocol implementations. In Bao, F., Li, H., and Wang, G., editors, *The 5th Information Security Practice and Experience Conference (ISPEC 2009)*, volume LNCS 5451, pages 171–182. Springer Berlin / Heidelberg. 12

Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. J., and Lear, E. (1996). Address Allocation for Private Internets. RFC 1918 (Best Current Practice). Available from: http://www.ietf.org/rfc/rfc1918.txt. 6