

MASARYK UNIVERSITY  
FACULTY OF INFORMATICS



# **Network Security Monitoring of Smart Home Systems**

PH.D. THESIS PROPOSAL

**Radek Krejčí**

Brno, January 2011

**Advisor:** prof. Ing. Václav Přenosil, CSc.

**Advisor's signature:** \_\_\_\_\_

## Contents

<b>Contents</b>	1
<b>1 Introduction</b>	2
<b>2 State of the Art</b>	5
2.1 <i>Network Security Monitoring</i>	5
2.1.1 <i>Flow Monitoring</i>	7
2.2 <i>SOHO and Building Automation Network Security</i>	9
2.2.1 <i>Communication in Building Automation Networks</i>	9
2.3 <i>Overview of the Threats</i>	11
2.3.1 <i>Man-In-The-Middle Attacks</i>	11
2.3.2 <i>PSYBOT</i>	13
2.3.3 <i>Chuck Norris Botnet</i>	14
2.3.4 <i>Stuxnet</i>	15
2.3.5 <i>Possible Consequences of Infected User Network Infrastructure</i>	16
<b>3 Aim of the Work</b>	18
3.1 <i>Research Fields</i>	18
3.1.1 <i>Network Behavior Analysis in a Context of a Smart Home Systems</i>	18
3.1.2 <i>Detection of Unexpected Equipment</i>	19
3.1.3 <i>Fingerprinting of Device Network Behaviour</i>	19
3.2 <i>Expected Results</i>	20
3.3 <i>Schedule of the Work</i>	20
<b>Bibliography</b>	24
<b>A Summary of My Present Work</b>	25
A.1 <i>Publications</i>	26
A.2 <i>Presentations</i>	27
A.3 <i>Posters</i>	27
A.4 <i>Participation in Projects</i>	27
A.5 <i>Teaching</i>	27
A.6 <i>Passed Courses</i>	27
<b>B Reprints of Selected Papers</b>	29

## Chapter 1

### Introduction

Computer networks became an essential part of the day-to-day activity in our modern lives. With growing number of security threats we gradually take security questions seriously into account. We install and use antivirus, antispam, antimalware and anti-whatever solutions that make fortresses from our PCs and make our data more secure. On the other hand Internet Service Providers (ISPs) improve the security of their networks. It starts with firewalls blocking malicious traffic. The next step of securing a network is made by Network-based Intrusion Detection/Prevention Systems (NIDS/NIPS).

The fact of growing cyber attacks and therefore increased importance the network security is also pointed out in the new Strategic Concept [1] of the North Atlantic Treaty Organization (NATO) adopted by its members' heads of state and government in 2010 in Lisbon. The concept defines today cyber attacks as more frequent, more organised and more costly in the damage they inflict. The defence against cyber attacks is declared as one of the NATO tasks in the next decade.

The main focus of nowadays methods and tools is to protect desktop and servers (host-based intrusion detection) and usually large high-speed ISPs' networks. These two areas are well protected mainly thanks to a fact that people now think of them as it is something that should be protected. But there is another part of connection between end user and service providing servers that stay underestimated in a security perspective. It is a user's Local Area Network (LAN) usually made of several Small Office/Home Office (SOHO) devices like cable and Asymmetric Digital Subscriber Line (ADSL) routers, network printers, heating or light system controllers, multimedia centers or shared network data storages (see Figure 1.1).

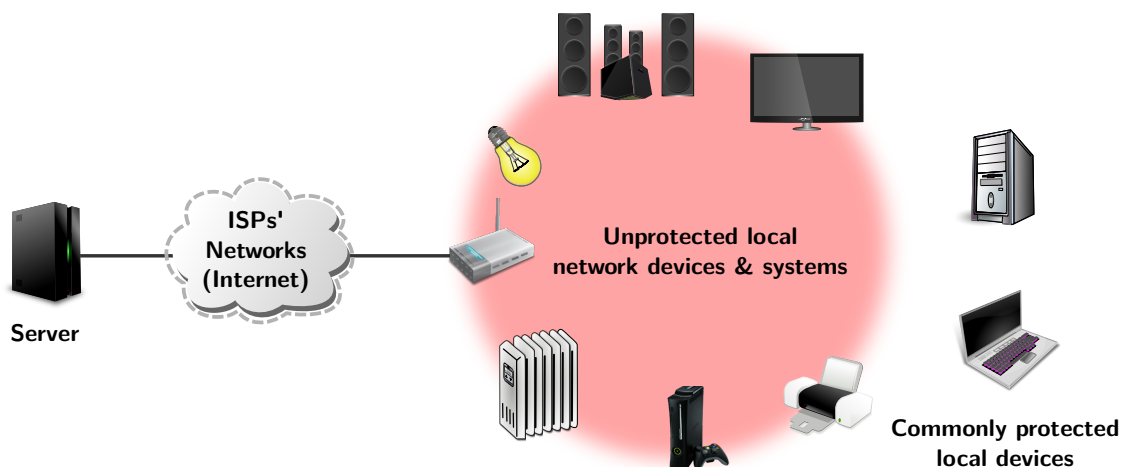


Figure 1.1: Path between user and server with highlighted unprotected parts.

Besides user security education or improving a passive security of a network devices, there is another way how to enhance a security of a network. Network Behavior Analysis (NBA) is one of NIDS/NIPS technologies used for the purposes of network security monitoring providing information about what is happening inside a network. Instead of packet inspection or signature detection, used by traditional NIDS/NIPS technologies, NBA watches a streams of related packets called flow<sup>1</sup>. This way NBA answers a questions about who is (or has been) communicating with whom, how, when, for how long and how much data is transferred.

Taking user network infrastructure under control has several significant advantages from the attacker's point of view. In a contrast to a user PCs, the most of a SOHO devices connected to a home network are not properly secured and configured to meet nowadays attacks. Users are not aware of securing *those little boxes that do something mysterious*. For users it is something that somebody – ISP's serviceman or someone who understand it – installed and since that the router, television, multimedia center, printer, heating system controller, etc. works out of the box. These devices are used as black-boxes that do not need any attention. But right this area is taking a key role in a smart home concept.

The concept of smart homes (e.g., [3] or [4]) with building automation and management systems is an incoming trend of a modern living. The smart homes are partially made of well known components, like wireless network access points, network printers or shared data stores, widely used in nowadays SOHO environments. But it newly connects these components together with automation and sensor networks better known rather from industrial networks. The security of a local computer networks connecting subsystems of the smart home is crucial for the whole system.

Unfortunately the security is a part of these systems that is not properly addressed so far. Partially it may be caused by a relative lack of interest from attackers. But some of a security holes originate in a push to decrease a cost of a SOHO devices and keep a development cheap, i.e. without dealing with a security. As predicted by Symantec [5] or by SANS Technology Institute [6], the upcoming generation of malware seems to be more specialized and narrowly targeted than before. This trend confirms the first examples of such malware described in the Section 2.3.

The possible change of upcoming attacks consists in its target. Attacker will focus instead of a protected desktops and servers on a less secured user network infrastructure. The most of possible malicious changes made by an attacker to the mentioned black-box devices stay completely invisible to common users. The possibility of securing PC is enabled by a broad range of antivirus, antispyware and other anti\* solution. But you have to be a quite advanced user with in-depth knowledge to secure your home router or network printer. Therefore the security of devices and thus a whole network depends on a default settings preset by a device vendor. So far the administrator's password is commonly set to a publicly known string, e.g., *admin*, *1234* or a blank password string. The situation would improve if vendors were generating random default administrator's passwords. This approach is already used by some ISPs for generating default Wired Equivalent Privacy (WEP) key used to connect to the SOHO Wi-Fi routers<sup>2</sup>. Information about such pre-generated password is then printed together with, e.g., MAC address, and a device is distributed with this unique information.

<sup>1</sup>A flow is defined as a set of packets passing an observation point in the network during a certain time interval. All packets belonging to a particular flow have a set of common properties (packet header field(s), characteristics of the packet itself or field(s) derived from packet treatment) [2].

<sup>2</sup>This approach is used, e.g., by Telefónica O<sub>2</sub> Czech Republic on their ADSL2+/ISDN routers.

Quite big advantage for an attacker is the fact that these devices are very often working 24 hours a day and 7 days a week. This is usually a difference to common desktop PCs that are better secured and that are not on-line all the time.

The intents of my Ph.D. thesis are to adapt current network security monitoring methods for the specific needs of the SOHO networks and interconnected automation and sensor networks. These methods will be used for security incident detection and network behavior monitoring. Prevention of the network security threats is out of scope of my Ph.D. thesis.

The Ph.D. thesis proposal is organised as follows. In the Chapter 2 we present the current state of the art. More detailed, we introduce technologies for the network security monitoring and present a current state of the security of SOHO and building automation networks and network devices. Then follows a short overview of the network security threats targeting these devices. In the Chapter 3 we propose aim of the work. The summary and the results of the previous work including reprints of the selected papers can be found in the Appendixes.

## Chapter 2

### State of the Art

The first part of the state of the art study introduces technologies for the network security monitoring. We focus on the Network Behavior Analysis with its flow monitoring approach. The second section addresses the security of devices and protocols used in SOHO and building automation networks. Finally we present a short overview of the network security threats targeted at SOHO networks.

#### 2.1 Network Security Monitoring

The process of a network security monitoring is performed by a wide range of devices belonging into the category of the Intrusion Detection System (IDS). Such devices are focused on identifying and reporting possible security incidents. We can divide IDS technologies into the four groups according to the types of events that they monitor [7].

**Network-Based** IDS is monitoring network traffic for a specific network segments or devices. It analyses the network, transport and application protocols operations to identify suspicious activity. Traditional network-based IDS inspects a packet payload to detect known threats.

**Wireless** IDS monitors and analyses wireless networking protocols to identify suspicious activity. This approach is not intended for monitoring higher-layer network protocols, e.g., Transmission Control Protocol (TCP) or User Datagram Protocol (UDP), transferred over a wireless network.

**Host-Based** IDS monitors events occurring within a single host. It gains information from a system logs, running applications and their activity and monitors file access and modifications or application configuration changes.

**Network Behavior Analysis (NBA)** monitors network traffic for an unusual traffic flows. Such traffic is usually generated by different types of attacks, such as (Distributed) Denial of Service (DoS/DDoS) attack, malware (e.g., worms or botnets) or network policy violations (like a workstation unexpectedly behaving like a server providing network services to other hosts). In a comparison to the traditional network-based IDS, NBA system uses statistical information about flows (number of packets, amount of transmitted data, used transfer protocol, etc.) instead of analysing a content of the transmission. This approach allows analysing of unencrypted as well as encrypted data in the same way.

The security is solved in particular level inside some network devices – network printers include simple firewalls or routers are accessible for management only from the local

network. But the security monitoring of a SOHO and building automation network is not targeted yet. We consider the NBA technology as the most promising approach for this task and we target it mainly in our future work.

IDS technologies use more different methods, usually together, to detect security threats. Generally they can be divided into the following three categories [7].

**Signature-Based Detection** is based on a comparison of the observed data with a patterns corresponding to a known threats. This method is very effective at detecting known threats with the static attack vector. On the other hand, signature-based detection is quite ineffective at detecting previously unknown threats or dynamically changing threats. Unfortunately with progressively growing amount of new versions and modifications of malware<sup>1</sup>, the signature-based detection becomes much more useless.

The typical example of the signature-based IDS is Snort<sup>2</sup> or its successor Suricata<sup>3</sup>.

**Stateful Protocol Analysis** is the process of comparing predetermined profiles of generally accepted definitions of benign protocol activities against observed events. The IDS must be capable of understanding and tracking the state of network, transport and application protocols. This approach is not widely used, mainly for the following reasons.

- We are unable to detect attacks that do not violate the generally acceptable protocol behavior, such as performing many benign actions in a short period of time to cause a denial of service.
- There are many differences between implementations and protocol specifications so used model can conflict with these implementation specifics.
- This approach has a high resource demands due to the complexity of the analysis. Overhead increases with tracking simultaneous sessions.

**Anomaly-Based Detection** follows the profile of a normal network traffic behavior and checks that the actually observed data correspond to the predefined profile. The profiles of normal activity include events connected to users, hosts, applications or statistics of network connections. The IDS uses statistical methods, e.g., a time series analysis known as Holt-Winters method, to compare the characteristics of current activity to thresholds related to the profile. The activity abnormalities are in most cases caused by typical malware activities such as sending large amount of emails (spam), making large number of connections or downloading data in an unusual way. The profiles are usually acquired by monitoring the behavior of the network over a period of time.

The process of building an initial profile is a common problem of anomaly-based detection. It is a challenging task to reflect real-world activity and prepare profile that will generate as a few false positives alerts as possible. There are two approaches to prepare a profile.

---

<sup>1</sup>Zeus, one of the most widespread botnet today, is provided with a builder toolkit to individualize its features according to an attacker requests. This way an attacker is able to create new Zeus clone with unique signature within a few minutes [8].

<sup>2</sup>[www.snort.org](http://www.snort.org)

<sup>3</sup>[www.openinfosecfoundation.org](http://www.openinfosecfoundation.org)



**Static profile** is one-time generated and unchanged until the IDS is directed to generate a completely new profile. The static profile will become inaccurate with changes of the system over time (growing number of users, changes in user needs, etc.) which leads to a production of false positive alerts.

**Dynamic profile** is adjusted continuously as new events are observed. Dynamic profiles are able to adapt to a changes of the system. But it also means, that they are susceptible to evasion attempts from attackers. For example, an attacker can perform small amounts of malicious activity in intervals, then gradually increase the frequency and quantity of the activity. If the rate of change is sufficiently slow, the IDS might think that the malicious activity is normal behavior and include it in its profile.

General problem of anomaly-based detection appears when an infected system is profiled. It makes the IDS think that a malicious activity is normal behavior of the system. In some cases, when administrators are able to detect and isolate such activity, it can be manually excluded from the profile.

To provide more accurate detection, IDS usually uses multiple detection methods.

### 2.1.1 Flow Monitoring

There are various systems for monitoring status of devices connected into the network. The best known and widely used are, e.g., Zabbix<sup>4</sup> or Nagios<sup>5</sup>. Design and implementation of such system focused on an automation control network devices using Building Automation and Control Networking (BACnet) protocol is described in [9]. This kind of monitoring systems gets information by active communication with the monitored devices, e.g., using Simple Network Management Protocol (SNMP), or with proprietary agents deployed on that devices.

Another approach to network monitoring is to gather information from network traffic, especially from the Internet Protocol (IP) flows. The flow monitoring serves as a main source of data for the NBA. Monitoring of the network flows was originally used for accounting/billing or network profiling and planning further development of the network. Over time it became a useful tool for the security incident handling and network forensics [10, Chapter 7]. There are two main possibilities to deploy flow monitoring device within a network [7].

**Inline** sensors are deployed so that the monitored network traffic must pass through the sensor. This type of the monitoring device is usually an integral part of some IDSs or other inline devices. Main motivation is to enable such device to directly stop attacks, detected by the flow monitoring. It can be done directly by blocking the suspicious network traffic passing through the device. Because the flow monitoring capability is usually presented as a special feature of devices (e.g., a router) used for other purposes, in a case of the device overload, e.g., as a result of the attack, the device starts to sample network traffic processed by a flow monitoring system. This decreases the load of the device, but it can distort results of the flow information analyses and network forensics.

---

<sup>4</sup>[www.zabbix.com](http://www.zabbix.com)

<sup>5</sup>[www.nagios.org](http://www.nagios.org)

**Passive** sensors monitor a copy of the network traffic. These devices are usually a standalone probes completely hidden to the attackers. This approach is more secured but less effective and flexible for purposes of intrusion prevention since it works with a traffic copy. In this case the prevention functionality is done by passing information about traffic to block into the active network device connected inline the network (kind of firewall in most cases).

In both described cases, gathered information is commonly sent to the center where the data are stored for further complex analyses. The center usually receives data from several probes deployed all over the network. Processing data from several points of the network provides overall outline of the network behavior. Furthermore, this information fulfils the demands of European Union<sup>6</sup> and Czech laws<sup>7</sup> for data retention. According to these laws, communication service providers must retain data generated or processed in connection with the provision of publicly available electronic communications services or of public communications networks for a specific time period (from 6 months to 2 years). The data can be then used by competent national authorities for the purposes of investigation, detection and prosecution of serious crimes.

A *de facto* standard for IP flow monitoring is Cisco NetFlow format [11]. While widely used NetFlow v5 format is static (only several items are gathered), NetFlow v9 uses dynamic format that allows administrators to select interested information for their specific purposes from the set of available items. For a unification of IP flow export protocols, Internet Engineering Task Force (IETF) proposed new standard, IP Flow Information Export (IPFIX) Protocol [2], with even a higher level of flexibility. Administrator is able to define its own items that will be observed in the network traffic.

Here we provide a summary of today widely used tools for the both flow acquisition and collection.

- **Aurora** is former IBM's research project of a traffic analysis and visualization system. The base AURORA system is now commercially available as Tivoli Netcool Performance Flow Analyzer. The system collects data in NetFlow or IPFIX format.  
Information available at [www.zurich.ibm.com/aurora/](http://www.zurich.ibm.com/aurora/)
- **fprobe** is a NetFlow probe based on the PCAP<sup>8</sup> library.  
Available from [fprobe.sourceforge.net](http://fprobe.sourceforge.net)
- **NFDUMP/NfSen** is a NetFlow collector with a graphical front end.  
Available from [nfdump.sourceforge.net](http://nfdump.sourceforge.net), [nfsen.sourceforge.net](http://nfsen.sourceforge.net)
- **ntop** is both a probe (called nProbe) and a collector tool supporting NetFlow protocols v5 and v9 as well as IPFIX protocol.  
Available from [www.ntop.org](http://www.ntop.org)
- **System for Internet-Level Knowledge (SiLK)** is a collection of traffic analysis tools developed by the CERT NetSA to facilitate security analysis of large networks. The SiLK tool suite supports the collection, storage, and analysis of network flow data.  
Available from [tools.netsa.cert.org/silk/](http://tools.netsa.cert.org/silk/)

<sup>6</sup>European Union Directive 2006/24/EC Regulation No. 127/2005 on electronic communication

<sup>7</sup>Act 127/2005 on Electronic Communications and on Amendment to Certain Related Acts.

<sup>8</sup>Packet CAPture library available at [www.tcpdump.org](http://www.tcpdump.org)

- **FlowMon** is a hardware-accelerated passive network monitoring probe [12] developed as part of the Liberouter<sup>9</sup> project held by CESNET<sup>10</sup> in cooperation with Masaryk university and University of Technology Brno. FlowMon is able to export data from the observation point in NetFlow v5, v9 and IPFIX format. The Flowmon probe is currently commercially available by INVEA-TECH spin-off company.

## 2.2 SOHO and Building Automation Network Security

SOHO devices are much easier to compromise than modern desktop or server systems, as illustrate research activities of GNUCITIZEN<sup>11</sup> [13]. Security threats disclosed by GNUCITIZEN researchers include, e.g., security weaknesses of used protocols (SNMP injection [14] or enabled Universal Plug and Play (UPnP) protocol [15]) or vulnerabilities of device management interface (e.g., authentication process bypass, privilege escalation or cross-site scripting (XSS)<sup>12</sup> vulnerability).

The main problem of such devices is a lack of security attention to these devices since they are *“just a primitive single purpose devices”*. Many attacks derive benefit from leaving SOHO devices with a factory settings including default password. There are several web sites providing overview of the default credentials for network devices from various vendors<sup>13</sup>.

Many networks stand, in the security point of view, mainly on the principles of a physical network security. But in such cases, other security mechanisms are usually omitted and if somebody is able physically access a device/link, he is able to do whatever he wants. On the other hand, typical sensor networks, as part of the building automation system, at least particularly operates in unfriendly outdoor environment where a physical attack is highly probable [16, Chapter 17]. For these cases a system that monitors changes in the network can be very profitable to allow a detection of enemy device in the network or unexpected changes in behavior of devices.

### 2.2.1 Communication in Building Automation Networks

Communication protocols used in a building automation systems were originally developed for use mainly on dedicated links (serial links in most cases) so security of transmitted data is not generally targeted by these protocols. With their shift to the open networks, administrators have been facing up to the needs of securing transported data.

There are some new protocols targeting security in automation and sensor networks, e.g., Perrig et al. [17] introduce SPIN – a suite of security blocks:

- **Secure Network Encryption Protocol (SNEP)** includes data confidentiality, two-party data authentication and evidence of data freshness.
- **Micro version of Timed, Efficient, Streaming, Loss-tolerant Authentication ( $\mu$ TESLA) Protocol** providing authenticated broadcast for severely resource-constrained environments.

<sup>9</sup>[www.liberrouter.org](http://www.liberrouter.org)

<sup>10</sup>Czech national research and educational network operator.

<sup>11</sup>Information security think tank, for more information see [www.gnucitizen.org](http://www.gnucitizen.org).

<sup>12</sup>Computer security vulnerability of web applications that enables malicious attackers to inject client-side script into web pages viewed by other users. (Wikipedia)

<sup>13</sup>[www.default-password.info](http://www.default-password.info) or [www.phenoelit-us.org/dpl/dpl.html](http://www.phenoelit-us.org/dpl/dpl.html).

Another approach is presented, e.g., by Honeywell and Byres Security Inc. providing commercial solutions to secure Modbus protocol (more described below) used on TCP networks [18]. They provide kind of firewall checking Modbus protocol for suspicious commands and responses.

There are three main communication protocols addressing the largest market of building automation networks [19]. BACnet seems to be the strongest one, proposed as the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE), American National Standards Institute (ANSI) and International Organization for Standardization (ISO) standard. LonWorks has the largest product base while Modbus is the cheapest one with a long history.

We will focus on BACnet in our future research due to its usage in the campus technological network at Masaryk University, which will serve as a testing environment for the monitoring tools being developed.

Here we provide just a short overview of mentioned protocols.

#### Modbus<sup>14</sup>

Modbus was published in 1979 and became a *de facto* standard communication protocol for industrial electronic devices connected over serial links. It is openly published and royalty free protocol. Besides using Modbus over serial links, there is also possibility to use it in the Ethernet networks as an application-layer protocol over TCP transport protocol. Modbus was originally designed to be used on dedicated lines. The problem of Modbus in networks is that it doesn't provide any mechanism to secure transferred information or to authenticate the sender – devices do whatever is told them to do by anyone [20].

#### LonWorks<sup>15</sup>

The LonWorks, developed in early nineties by Echelon Corp., represents the complete technology platform for communication in control networks. The communication protocol of the LonWorks platform, called LonTalk, was accepted as ANSI<sup>16</sup> standard control networking protocol in 1999. LonTalk uses peer-to-peer communication architecture to pass messages between devices. LonTalk doesn't provide data encryption but the sender authentication mechanism is optionally (it doubles amount of transferred data) available.

#### BACnet<sup>17</sup>

BACnet is a communication protocol designed in early nineties to connect building automation and control systems, such as lighting, heating or access control and their associated equipment. The protocol became ASHRAE and ANSI standard<sup>18</sup> in 1995, ISO standard<sup>19</sup> in 2003 and currently it is a national standard in more than 30 countries. BACnet is based on a client-server communication model. Device functions are accessed as objects with defined properties (according to a device class).

<sup>14</sup>[www.modbus.org/specs.php](http://www.modbus.org/specs.php)

<sup>15</sup>[www.echelon.com/developers/lonworks/](http://www.echelon.com/developers/lonworks/)

<sup>16</sup>ANSI/CEA-709.1-B

<sup>17</sup>[www.bacnet.org](http://www.bacnet.org)

<sup>18</sup>ASHRAE/ANSI Standard 135

<sup>19</sup>ISO 16484-5:2003

There are 5 networking technologies that can be used to connect BACnet devices. BACnet also defines how to connect sub-networks of different types together via routers.

- **Ethernet**, standardized as IEEE standard 802.3, is a networking technology for LANs. Ethernet can be used directly (BACnet-over-Ethernet) or together with the higher level IP network protocol (BACnet-over-IP).
- **ARCNET** is a LAN protocol used similarly to the Ethernet.
- **Master-Slave/Token-Passing (MS/TP)** for devices with lower speed requirements – MS/TP network is designed to run at speeds of 1 Mbps or less over twisted pair wiring.
- **LonTalk** protocol mentioned above.
- **Point-To-Point (PTP)** protocol is used over RS-232<sup>20</sup>.

The dominant technology used in the campus technological network at Masaryk University is BACnet-over-Ethernet [9, p. 7]. Also MS/TP technology is used to connect controllers within a single campus pavilion and BACnet-over-IP is used to connect controllers placed in different sub-networks, e.g., to connect controllers in the campus Bohunice with controllers located in Faculty of informatics on Botanická street.

### 2.3 Overview of the Threats

In this section we describe real-world security threats targeting an area of SOHO and automation system networks. The first part introduces Man-In-The-Middle (MITM) attacks in general. Also description of several attack vectors using advantages of the MITM position is included, because a compromised SOHO network infrastructure is an ideal place for this type of attacks.

Then, as a reference to a generation of malicious software targeting technological and SOHO network infrastructure, we present a short overview of such recently detected malware. PSYBOT was detected in 2009 and it targets cable/ADSL modems and routers used in a SOHO networks. At the end of year 2009, the Chuck Norris botnet appeared as a kind of successor of the PSYBOT targeting the same type of devices. The Stuxnet worm was detected in 2010 targeting industrial networks (namely Supervisory Control And Data Acquisition (SCADA) systems [21]). The Stuxnet is currently designated as the most sophisticated worm in computer history.

#### 2.3.1 Man-In-The-Middle Attacks

The MITM attack is a form of active eavesdropping in which an attacker is able to make two victims (user and server) believe that they are communicating directly to each other, when in fact the connection is controlled by the attacker. The attack may be used simply to gain access to the transmitted data or to modify the messages before retransmitting them.

Compromised devices of a user network infrastructure are a perfect place for execution of a MITM attacks. Especially SOHO routers (or ADSL modems) are nice targets for attackers since the most of the network traffic in SOHO network is passing through this device. When

<sup>20</sup>EIA Standard RS-232-C Interface Between Data Terminal Equipment and Data Communication Equipment Employing Serial Data Interchange

attacker is able to take control of such device, performing some kind of a malicious operation to capture or modify user data is quite simple. Even if some kind of secured protocol, e.g., Secure Socket Layer or Transport Layer Security (SSL/TLS), is used, there are still an effective ways how to get into the communication session [22, 23]. Very often the most complicated part of these attacks is to get into the man-in-the-middle position.

The following paragraphs describe three example scenarios of using compromised SOHO router for damaging purposes.

### DNS Spoofing

The most SOHO routers are configured to provide auto configuration of IP settings by the Dynamic Host Configuration protocol (DHCP). In addition to setting up IP addresses, DHCP also provides further configuration data as an addresses of a Domain Name System<sup>21</sup> (DNS) resolvers.

Compromised router can start spoofing (specific) DNS queries. The replies contain a spurious IP address of requested domain name and user starts a communication with someone else than required. In most cases this attack stays completely invisible for users.

The solution of the problem may be achieved by using the Domain Name System Security Extensions (DNSSEC) [24] by both client and server.

### Certificates Forgery

In a context of protocols using SSL/TLS as underlying protocols, e.g., Hypertext Transfer Protocol Secure (HTTPS), impersonating an inappropriate domain name (DNS spoofing) should be avoided using server certificates. Certificates identify the server. The certificate authenticity is guaranteed by a digital signature of a trustworthy Certification Authority (CA). Besides the fact that due to many false positives warnings (thanks to misusing the certificates or using untrustworthy CA by server operators) and therefore ignoring the most of warnings by users [25, 26], there are also possibilities to forge a certificate by either an attacker's server or directly by the MITM.

There is a known issue of certificates using hashes produced by Message-Digest algorithm 5 (MD5). Demonstration of such attack is described in [27]. The attack takes advantage of a weakness in the MD5 cryptographic hash function allowing the construction of different messages with the same MD5 hash (a.k.a. MD5 collision vulnerability). This vulnerability enables attacker to create a rogue CA with certificate trusted by clients (browsers, email clients, etc.).

Another possibility how to forge a certificate is null-prefix vulnerability. Moxie Marlinspike disclosed this vulnerability at the Black Hat DC 2009 conference [28]. The issue lies in implementation flaws of Microsoft's Crypto API, Mozilla's NSS and other libraries providing functions to validate SSL/TLS certificates. The problem is in a mistaken interpretation of the null ('\0') character. As a result of the attack, affected client programs validate the certificate for a different domain name than the one checked by CA.

Both mentioned vulnerabilities (and many others) are implemented by *sslsniff*<sup>22</sup> tool and can be used by anyone with basic networking skills.

<sup>21</sup>The system for translation of human readable domain names into numerical identifiers associated with networking equipment for the purpose of addressing these devices worldwide.

<sup>22</sup>[www.thoughtcrime.org/software/sslsniff/](http://www.thoughtcrime.org/software/sslsniff/)

## SSL Strip

The SSL Strip attack was introduced by Moxie Marlinspike at the Black Hat DC 2009 conference [29]. The attack benefits from the way how users often browse the web pages and access secure servers. They are often redirected to a secured protocol from unsecured web pages or follow hypertext links. In these cases the attacker replaces secured links and alters redirection headers to keep user communication unencrypted. MITM then communicates with the server in a secure way and the server itself is not able to detect anything wrong. But all information from the user is compromised due to unsecured communication with the MITM sitting inside the infected SOHO device.

The proof of concept implementation of this attack can be found as *sslstrip*<sup>23</sup> tool. There is no simple way how to generally prevent this attack. A partial improvement can be achieved by using Strict-Transport-Security [30] (STS) mechanism on both client and server. But it is not able to prevent SSL Strip attack when the site is accessed for the first time.

### 2.3.2 PSYBOT

The PSYBOT was the first worm targeting SOHO devices to create a botnet of them. It exploited different vulnerabilities and misconfiguration to infect the SOHO devices, namely ADSL modems, and then continue spreading the infection to other vulnerable devices.

The first discovered version of the PSYBOT bot (PSYBOT 2.5L) was described by Australian security researcher Terry Baume [31] in January 2009. Terry detected the bot at his Netcomm NB5 ADSL modem with the MIPS architecture and an embedded Linux distribution (BusyBox<sup>24</sup> with MontaVista Linux 2.4.17 kernel). The following version (PSYBOT 2.9L) attracted attention in March 2009, when the DroneBL site<sup>25</sup> was a target of a DDoS attack [32]. The source of the attack was identified as the PSYBOT's enslaved devices. After the attack against DroneBL the botnet was shut down by its master who pronounced that it was a fun but a "research" is over.

The size of the botnet was about 80-100 thousands of bots as estimated DroneBL and declared by the botnet master.

Target vector was quite simple because a several revisions of the modem firmware was shipped with web configuration interface available from the WAN and disabled authentication – no username or password was required to access the configuration interface. Also access via Secure Shell (SSH) and Telnet was enabled with root password set by default to the 'admin' string. These flaws were corrected in later firmware revisions.

The mechanism of infection and spreading the botnet was as follows:

- Connect to a modem via Telnet.
- Try a default password and then spawn a shell.
- Download a botnet binaries and execute them.
- Reject all other connections to the configuration interface.
- Connect to the Internet Relay Chat (IRC) Command & Control (C&C) server and wait for orders.

<sup>23</sup>[www.thoughtcrime.org/software/sslstrip/](http://www.thoughtcrime.org/software/sslstrip/)

<sup>24</sup>[www.busybox.net](http://www.busybox.net)

<sup>25</sup>DroneBL monitors the abused IPs and provides a lists of them to be used for e.g., blacklisting.

The disinfection of the device is quite simple when a user detects the presence of the bot. The device's firmware is stored in a kind of persistent Non-Volatile Random Access Memory (NVRAM) and the bot doesn't affect firmware by any malicious upgrade. The PSYBOT resists in Random Access Memory (RAM) used by embedded devices as a temporary extension storage that is erased with each reboot. Power cycling the device disinfects the device but that doesn't prevent any future infection until a security configuration of the device is changed.

The PSYBOT is frequently presented as the first malware targeting SOHO devices, especially modems and routers, at all. But the first worm utilizing SOHO routers was Coldbot in 2003. Coldbot was a bot infecting PCs running Windows operating systems. But for connecting to its IRC C&C server the Coldbot utilized a set of compromised routers as proxies. This way the Coldbot was hiding its presence in the network.

### 2.3.3 Chuck Norris Botnet

We have discovered this botnet at the beginning of December 2009. We call the botnet after Chuck Norris because an early version included the following string.

```
[R]anger Killato : in nome di Chuck Norris !
```

In previous years we have developed and deployed our own NetFlow-based network monitoring system. This system is currently used to anomaly detection and security analyses on the Masaryk university network. At the beginning of the December 2009 the system showed an unusual amount of Telnet scans. Tracing back to a sources of these scans we have identified world wide infected ADSL modems and SOHO routers.

The Chuck Norris botnet targets, similarly to PSYBOT, SOHO devices built on MIPSel architecture running Linux kernel. It obtains orders from its IRC C&C centers. According to our analysis [33], the IRC servers appeared to public as a porn sites. But for the botnet purposes they were providing hidden directories with updates of the botnet binaries.

The lifetime of the Chuck Norris botnet can be divided into the following four parts.

1. Scan for a vulnerable devices in a selected networks. Bot contains a list of network segments belonging to broadband Internet providers deploying targeted SOHO devices to its customers in a large scale. Examples of the network segments and their owners are shown in Table 2.1.

IP Prefix	Owner
217.236.0.0/16	Deutsche Telekom
194.206.0.0/16	France Telecom
213.98.0.0/16	Telefonica de Espana
88.253.0.0/16	TurkTelekom
87.22.0.0/16	Telecom Italia
201.1.0.0/16	Telecomunicacoes de Sao Paulo

Table 2.1: Example of IP prefixes encoded in the Chuck Norris botnet binaries.

2. The bot performs a Telnet brute force attack against vulnerable devices detected by scans during the first phase. In a comparison to PSYBOT, the Chuck Norris botnet utilize only 15 combinations of a login and password. Used combination can be found in the Table 2.2.



User	Password
root	admin, Admin, password, root, 1234, private, XA1bac0MX, adsl1234, %%fuckinside%%, dreambox, <i>blank password</i>
admin	admin, password, <i>blank password</i>
1234	1234Admin

Table 2.2: Default passwords used for a dictionary attack to compromise a Telnet service.

3. The bot initialize itself when connecting to the IRC C&C center and reading and interpreting the topic of IRC channel as an initial command. This command is usually used to update bot binaries. The bot also blocks remote connections to the device's TCP ports 22-80 to disable access for other bots and administrators.
4. During the last stage, installed bot performs further scanning for vulnerable devices and waits for the attack commands from the IRC C&C.

To obtain this knowledge we have prepared vulnerable MIPSel router and infected it. We have been recording all incoming and outgoing data until the botnet stopped the activity on February 23<sup>rd</sup>, 2010. It happened shortly after the information about the Chuck Norris botnet was publicly presented on February 15<sup>th</sup> [34]. During the botnet monitoring, we have observed several occurrences of real DDoS attacks and DNS spoofing attacks (see Section 2.3.1).

In the same way and due to same reasons as in the case of the PSYBOT, disinfection of the infected device can be simply done by power cycling.

On bases of our analyses, the Network Security Department of the Institute of Computer Science (ICS) at Masaryk university developed Chuck Norris detection plug-in for the Nf-Sen flow collector. The plug-in is available<sup>26</sup> as one of the results of the Computer Incident Response Capability Development in the Cyber Defence Environment (CYBER)<sup>27</sup>.

### 2.3.4 Stuxnet

Stuxnet is an example of an incoming trend in malware – highly precise targeted worm attacking the specific type of devices. In this case it concerns the industrial SCADA systems. Despite a similar attack against smart home systems is not publicly known so far, it is obvious that this kind of a closely targeted malware will be more common in a near future. A malware specialization to smart home systems is highly probable with widening of such systems.

Stuxnet was detected in July 2010. But an early version of the Stuxnet was dated back to March 2009. Since then the worm went through a continuous development.

Stuxnet takes advantage of (at least) four zero-day<sup>28</sup> vulnerabilities of the Windows operating systems and other vulnerabilities of the Siemens SIMATIC WinCC, PCS7 and S7

<sup>26</sup> Available from [www.muni.cz/ics/research/cyber/chuck\\_norris\\_botnet](http://www.muni.cz/ics/research/cyber/chuck_norris_botnet).

<sup>27</sup> CYBER Project website is available at [www.muni.cz/ics/research/cyber/](http://www.muni.cz/ics/research/cyber/).

<sup>28</sup> A zero-day vulnerability is unknown to others or there is not still any fix or patch available in a time of the attack.

product lines. It uses at least 7 propagation mechanisms to spread itself to other computers. These mechanisms can be divided into the following three groups.

- Propagation via removable Universal Serial Bus (USB) devices.
- Propagation via network communication.
- Propagation via Siemens project files.

Although the Stuxnet can infect any Windows-based system, it performs further malicious actions only to SCADA systems using Siemens Programmable Logic Controllers (PLCs). Stuxnet detects and then modifies the specific types of Siemens PLCs. If such PLC is found, Stuxnet reprograms several blocks of PLC. The modified PLC then waits for a specific event. If the event occurs the executing process of an original logic is changed with possible destructive results. But real impact to industrial processes is unknown to public. Based on reports from industrial customers, Siemens announced only 22 control systems that have been affected by Stuxnet (as of November 22, 2010) [35]. According to Symantec report [36], the most Stuxnet occurrence was detected in Iran, Indonesia and India.

Besides changes in PLCs, Stuxnet is also able to control communication between PLC and a Siemens programming station to hide changes made to PLC. In addition, the Stuxnet binaries are signed by trusted private certificates. These certificates were stolen (in an unknown way) from Realtec Semiconductors and JMicron. The stolen certificates were then revoked, but already signed executables will be still running correctly.

Due to all mentioned characteristics (detailed description can be found in [37]), Stuxnet is considered to be one of the most complex and sophisticatedly engineered worms.

### 2.3.5 Possible Consequences of Infected User Network Infrastructure

Besides attackers' advantages, mentioned earlier in the Introduction, compromising a SOHO network infrastructure has an important consequences for users.

Despite of negative results of the user PC antivirus scans, the ISP still detects a malicious behaviour of connected user's network. According to specific ISP's conditions, such user can be disconnected or sanctioned in some other way.

Keeping malware invisible gives a user false feeling of security. But all outgoing information from a user PCs is compromised including:

- bank account credentials,
- credit card information,
- personal identity information,
- or private documents printed at local network printer.

The detailed description of a malware, especially botnet, economy principles can be found in [38].

As shown by Moxie Marlinspike [29], usage of secured protocols (such as SSL/TLS) is not enough in the case of an advanced MITM attacks. Modems and routers working as an

Internet gateway are great places for such purposes, e.g., phishing<sup>29</sup> is essentially much more successful when it is combined with DNS spoofing attack [39], that can be very often simply done by an infected local network gateway.

Besides the lost of privacy, users usually lose their resources – computing power of the devices but more often a capacity of the network connection line, because the most of nowadays bots and other enslaved devices are used to send a spam.

---

<sup>29</sup>The criminally fraudulent process of attempting to acquire sensitive information by masquerading as a trustworthy entity in an electronic communication. (Wikipedia)

## Chapter 3

### Aim of the Work

The aim of the presented work is to modify network security monitoring methods, especially NBA, used in large and high-speed networks for the area of a local home networks interconnected with the building automation and sensor networks used for purposes of smart home systems. In a network monitoring point of view this type of network has a specific needs in a sense of limited resources of the network devices or used communication protocols. Furthermore, we are going to design and evaluate new detection methods combining the network monitoring of ISP's network with the home network monitoring of connected user networks. We plan to use these methods and developed tools for a detection of malicious software (like viruses, worms or botnets) and compromised devices.

### 3.1 Research Fields

#### 3.1.1 Network Behavior Analysis in a Context of a Smart Home Systems

The NBA became an important way how to enhance the network security of a large multi-gigabit networks. IP flow monitoring is provided by either standalone passive monitoring probes or by active network devices (e.g., routers) as their additional feature. The NBA gives an overview of what is happening inside the network collecting information from many observation points.

We believe that the usage of NBA approach can be highly profitable in an environment of a smart home system networks. As well as in large-scale networks, the flow monitoring provides information about changes of the device behavior. Furthermore, with less data passing through the local network we plan to collect more detailed network traffic information for further analysis. This will be enabled by using flexible flow export protocols, especially IPFIX. The approach of collecting specific, and in some cases newly defined (in a context of IPFIX), information will be needed also in a case of the building automation and sensor networks. There are specific communication protocols used inside these types of networks. For purposes of a flow monitoring inside this environment we have to modify a definition of a flow and its key information items in contrast to predominantly used IP flow definition. The identification of a useful key items describing communication flow between two objects in the network must be done with an awareness of combining observed data from slightly different environment. The most of data are transferred via Ethernet with IP protocol. In this case information from the IP (eventually TCP/UDP) header is sufficient. But in a case of sensors and control devices in a building automation networks using a specific communication protocols like BACnet, Modbus, M-Bus, etc., necessary information must be acquired from the higher protocol headers. For these purposes we suppose to base our tools on know-how gained at CESNET's Liberouter project during the development of the FlowMon software.

Selected possibilities of using collected data are described in following paragraphs and they will shape my research.

### 3.1.2 Detection of Unexpected Equipment

After discovery of the Chuck Norris botnet we have been investigating possibilities of using infected router as a tool for MITM attacks against SSL/TLS protocols (see summary of my present work in Appendix A). Besides other threats we have highlighted the attack vector presented by Moxie Marlinspike at Black Hat DC 2009 – SSL Strip [29]. It operates like a transparent proxy. Requested secured connection is divided into secured connection between the proxy and server and unsecured connection between the proxy and user. Data passing through the unsecured connection are completely visible for the attacker.

Currently I supervise a master's thesis focused on transparent proxy detection. Similarly to a Network Address Translation (NAT) detection flow-based methods [40], we are going to detect transparent proxies according to a flow information. It can be achieved by comparing parameters of different types of the traffic. Transparent proxy usually affects only a specific network traffic, e.g., SSL Strip (Section 2.3.1) modifies only HTTP traffic. Parameters of HTTP flows then will differ in Time To Live (TTL) item of IP headers from the TTL item in other network traffic flows.

Another examples of equipment, which presence can point to some kind of a security threat are devices behaving like a servers providing a services (data storage, mail server, etc.), newly (and unexpectedly) connected device in a sensor network and others. The identification and detection of such equipment will be part of my following research.

Generally there are two possible approaches for the equipment detection.

1. Passive monitoring based on flow information.
2. Active probing of the network and interconnected devices.

We suppose a cooperation of the both approaches according to a specific type of the equipment that we detect. Proposed methods and their implementation are supposed to be part of the research results.

### 3.1.3 Fingerprinting of Device Network Behaviour

The previous point of interest is closely connected to a fingerprinting of device network behaviour. Fingerprints of a benign device behavior can be used for a detection of a malicious equipment deployed inside the infected device. Since we plan to use flow-based approach for a detection of unexpected equipment inside a network, we need fingerprints of network behaviour visible from flow information of such devices.

Besides the detection of unexpected devices, with flow-based fingerprints we will be able to detect changes in behaviour of standard devices (e.g., network printers). This way we can automatically identify infected or compromised devices inside a network.

Real-world experiences from the ICS's Network Security Department indicate that administrators are very interested in a behavior of new devices connected into their network. It would be useful to have benign behavior profiles of that devices.

In addition, fingerprints of the device behavior and therefore fingerprints of the specific network protocol implementations can contribute to verification of their resistance against side channels attacks<sup>1</sup>.

---

<sup>1</sup>Side channel attack is any attack based on information gained from the implementation or behavior of a system, rather than brute force or theoretical weaknesses in the algorithms and protocols. (Wikipedia)

### 3.2 Expected Results

The expected output of my Ph.D. thesis will be:

- Study of usability and deployment possibilities of network security detection methods known from high-speed networks within smart home system networks. We will be also interested in a cooperation of such tools deployed in the ISP and SOHO networks. The aim of my interest is at behavioral analysis of network traffic.
- Modification of current IP flow monitoring tools for specific needs of the technological network. Data must be collected and further analysis must be done based on information observed from protocols like Ethernet or BACnet.
- Tools and methods for malware and unexpected equipment detection in SOHO devices and networks. The target of detection tools includes worms or botnets as well as transparent proxies or unexpected device connected into a technological network.
- Study of modified tools capabilities to detect malware infecting a smart home infrastructure.
- Fingerprints of benign network behavior of devices used in local networks and smart home systems (network printers, routers/modems, IP cameras, etc.).

### 3.3 Schedule of the Work

My rough study and research plan is as follows.

#### **Spring 2011**

Modification of the IP flow monitoring tools according to specific needs of the campus technological network at Masaryk University. State doctoral exam end defence of this Ph.D. thesis proposal.

#### **Autumn 2011**

Fingerprinting of benign and malicious behavior SOHO devices (printers, IP cameras, smart home system controllers, etc.). Implementation of methods for the transparent proxy detection.

#### **Spring 2012**

Identification and detection of unexpected devices in SOHO and automation system networks.

#### **Autumn 2012**

Implementation and usability testing of promising methods for network security within smart home system networks.

#### **Spring 2013**

Deployment of an acquired knowledge and developed tools in the campus technological network and local networks within Masaryk University (continual task during my whole study). The Ph.D. thesis defence.

## Bibliography

- [1] NATO. Active Engagement, Modern Defence: Strategic Concept For the Defence and Security of The Members of the North Atlantic Treaty Organisation, November 2010. [cited 2010-12-14]. Available from: <http://www.nato.int/lisbon2010/strategic-concept-2010-eng.pdf>.
- [2] B. Claise. Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information. RFC 5101 (Proposed Standard), January 2008. Available from: <http://www.ietf.org/rfc/rfc5101.txt>.
- [3] e-Home AUTOMATION. e-Home AUTOMATION website [online]. Available from: <http://www.e-homeautomation.com/> [cited 2010-12-15].
- [4] Smart Home Systems. Smart Home Systems website [online]. Available from: <http://www.smarthomesystems.com/> [cited 2010-12-15].
- [5] Kevin Haley. Internet Security Predictions for 2011: The Shape of Things to Come, December 2010. [cited 2010-12-19]. Available from: <http://www.symantec.com/connect/blogs/internet-security-predictions-2011-shape-things-come>.
- [6] Josh Wright. Security Predictions 2011 & 2012 – The Emerging Security Threat, 2010. [cited 2010-12-19]. Available from: [http://www.sans.edu/resources/securitylab/security\\_predict2011.php](http://www.sans.edu/resources/securitylab/security_predict2011.php).
- [7] Karen Scarfone and Peter Mell. Guide to Intrusion Detection and Prevention Systems (IDPS), February 2007. Recommendations of the National Institute of Standards and Technology. Available from: <http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf>.
- [8] Sourcefire Vulnerability Research Team. Zeus Trojan Analysis [online]. Available from: <http://labs.snort.org/papers/zeus.html> [cited 2011-01-04].
- [9] Adam Kučera. Monitorovací nástroje pro objekty a zařízení sítě BACnet, 2010. Bakalářská práce (česky), Fakulta informatiky Masarykovi univerzity. Available from: [https://is.muni.cz/auth/th/255658/fi\\_b/](https://is.muni.cz/auth/th/255658/fi_b/).
- [10] Radek Krejčí. Network Traffic Collection with IPFIX Protocol. Master's thesis, Masaryk university, Faculty of informatics, 2009. Available from: <http://theses.cz/id/19mzlh/>.
- [11] B. Claise. Cisco Systems NetFlow Services Export Version 9. RFC 3954 (Informational), October 2004. Available from: <http://www.ietf.org/rfc/rfc3954.txt>.

- 
- [12] Martin Žádník, Libor Polčák, Ondřej Lengál, Martin Elich, and Petr Kramoliš. FlowMon for Network Monitoring, 2010. Technical report. Available from: <http://www.cesnet.cz/doc/techzpravy/2010/flowmon/>.
  - [13] Adrian Pastor. Cracking into Embedded Devices and Beyond. Presented at the Hack in the Box conference in Dubai on April 24, 2008. Available from: <http://www.gnucitizen.org/static/blog/2008/04/cracking-into-embedded-devices-hitb-dubai-2008.pdf>.
  - [14] Adrian Pastor. SNMP Injection – Achieving Persistent HTML Injection via SNMP on Embedded Devices, 2008. Available from: [http://www.procheckup.com/vulnerability\\_manager/documents/document\\_1258758662/SNMP\\_injection.pdf](http://www.procheckup.com/vulnerability_manager/documents/document_1258758662/SNMP_injection.pdf).
  - [15] Ryan Giobbi. UPnP enabled by default in multiple devices, 2008. Available from: <http://www.kb.cert.org/vuls/id/347812>.
  - [16] Yang Xiao. *Security in Distributed, Grid, Mobile, and Pervasive Computing*. Auerbach Publications, Boston, MA, USA, 2007.
  - [17] Adrian Perrig, Robert Szewczyk, J. D. Tygar, Victor Wen, and David E. Culler. SPINS: security protocols for sensor networks. *Wirel. Netw.*, 8:521–534, September 2002. Available from: <http://dx.doi.org/10.1023/A:1016598314198>.
  - [18] Tofino Security. *Honeywell Modbus TCP Firewall*. Available from: <http://www.tofinosecurity.com/sites/default/files/EPS9211-ET-HN1-2.pdf>.
  - [19] Nino Kurtalj. Protocol Positioning from Field to Cloud. *AutomatedBuildings.com*, October 2010. Available from: <http://www.automatedbuildings.com/news/oct10/articles/brightcore/100928023606brightcore.htm>.
  - [20] Adrien de Beaupre. Cyber Security Awareness Month – Day 22 port 502 TCP – Modbus, 2009. [cited 2011-01-09]. Available from: <http://isc.sans.edu/diary.html?storyid=7426>.
  - [21] Wikipedia. Scada — Wikipedia, the free encyclopedia, 2010. [cited 2010-12-15]. Available from: <http://en.wikipedia.org/wiki/SCADA>.
  - [22] Radek Krejčí and Pavel Čeleda. (Ne)bezpečné HTTPS - část I. *Data Security Management*, 14(3):38–41, 2010.
  - [23] Radek Krejčí and Pavel Čeleda. (Ne)bezpečné HTTPS - část II. *Data Security Management*, 14(4):22–27, 2010.
  - [24] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. DNS Security Introduction and Requirements. RFC 4033 (Proposed Standard), March 2005. Updated by RFC 6014. Available from: <http://www.ietf.org/rfc/rfc4033.txt>.
  - [25] Joshua Sunshine, Serge Egelman, Hazim Almuhiemedi, Neha Atri, and Lorrie Faith Cranor. Crying wolf: an empirical study of SSL warning effectiveness. In *Proceedings of the 18th conference on USENIX security symposium, SSYM'09*, pages 399–416, Berkeley, CA, USA, 2009. USENIX Association. Available from: <http://portal.acm.org/citation.cfm?id=1855768.1855793>.



- [26] Rachna Dhamija, J. D. Tygar, and Marti Hearst. Why phishing works. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, pages 581–590, New York, NY, USA, 2006. ACM. Available from: <http://doi.acm.org/10.1145/1124772.1124861>.
- [27] Alexander Sotirov, Mar Stevens, Jacob Appelbaum, Arjen Lenstra, David Molnar, Dag Arne Osvik, and Benne de Weger. MD5 considered harmful today: Creating a rogue CA certificate. Presented at the 25th Chaos Communication Congress in Berlin on December 30, 2008. Available from: <http://www.win.tue.nl/hashclash/rogue-ca/>.
- [28] Moxie Marlinspike. Null Prefix Attacks Against SSL/TLS Certificates. Presented at the Black Hat DC Conference in Washington, DC on February 18, 2009. Available from: <http://www.thoughtcrime.org/papers/null-prefix-attacks.pdf>.
- [29] Moxie Marlinspike. New Tricks For Defeating SSL/TLS. Presented at the Black Hat DC Conference in Washington, DC on February 18, 2009. Available from: <http://blackhat.com/html/bh-dc-09/bh-dc-09-archives.html#Marlinspike>.
- [30] J. Hodges, C. Jackson, and A. Barth. HTTP Strict Transport Security (HSTS). Internet Draft, July 2010. Available from: <http://tools.ietf.org/html/draft-hodges-strict-transport-sec-02>.
- [31] Terry Baume. PSYBOT Information Page [online]. Available from: <http://baume.id.au/psyb0t> [cited 2010-12-25].
- [32] Nenolod. Network Bluepill - stealth router-based botnet has been DDoSing dronebl for the last couple of weeks. [cited 2010-12-25]. Available from: <http://www.dronebl.org/blog/8>.
- [33] Pavel Čeleda, Radek Krejčí, Jan Vykopal, and Martin Drašar. Embedded Malware – An Analysis of the Chuck Norris Botnet. In *Proceedings of the 2010 European Conference on Computer Network Defense*, EC2ND '10, pages 3–10, Washington, DC, USA, 2010. IEEE Computer Society. Available from: <http://dx.doi.org/10.1109/EC2ND.2010.15>.
- [34] Čeští specialisté na kybernetickou obranu objevili nový kybernetický útok. Tisková zpráva. Available from: [http://www.army.cz/images/id\\_15001\\_16000/15609/023.doc](http://www.army.cz/images/id_15001_16000/15609/023.doc).
- [35] Siemens AG. SIMATIC WinCC/SIMATIC PCS 7: Information concerning Malware/Virus/Trojan. [cited 2010-12-27]. Available from: <http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&objid=43876783>.
- [36] Jarrad Shearer. W32.Stuxnet. [cited 2010-12-27]. Available from: [http://www.symantec.com/security\\_response/writeup.jsp?docid=2010-071400-3123-99](http://www.symantec.com/security_response/writeup.jsp?docid=2010-071400-3123-99).

- [37] Eric Byres and Scott Howard. Analysis of the Siemens WinCC/PCS7 "Stuxnet" Malware for Industrial Control System Professionals, October 2010. Available after registration. Available from: <http://www.tofinosecurity.com/professional/siemens-pcs7-wincc-malware>.
- [38] Yury Namestnikov. The economics of Botnets. *Securelist*, July 2009. Available from: <http://www.securelist.com/en/analysis?pubid=204792068>.
- [39] Tamara Dinev. Why spoofing is serious internet fraud. *Commun. ACM*, 49:76–82, October 2006. Available from: <http://doi.acm.org/10.1145/1164394.1164398>.
- [40] Vojtěch Krmíček, Jan Vykopal, and Radek. Krejčí. Netflow Based System for NAT Detection. In *Co-Next Student Workshop '09: Proceedings of the 5th international student workshop on Emerging networking experiments and technologies*, pages 23–24, New York, NY, USA, 2009.

## Appendix A

### Summary of My Present Work

During the past three semesters I have focused on a research in the area of SOHO devices security. In this phase, I was analysing especially weaknesses connected mainly to home wireless routers. In December 2009 I have participated on discovery of new botnet spreading to SOHO devices based on MIPSel architecture with the Linux kernel. The botnet was named Chuck Norris. Description of this botnet can be found in Section 2.3.3. The full story about detection and investigation of the Chuck Norris botnet can be found in the article *Embedded Malware – An Analysis of the Chuck Norris Botnet* presented at *European Conference on Computer Network Defense 2010*. The reprint of the article is available as part of Appendix B.

After presenting the threat of the Chuck Norris botnet publicly, I have focused on possible further usages of a malware similar to Chuck Norris – malware attacking user network infrastructure. I did a research of known techniques used to overhear and decrypt or modify network traffic secured with SSL/TLS protocols. The results are summarized in two articles *(Ne)bezpečné HTTPS* (in Czech) published in reviewed journal *Data Security Management*. Mainly the threat of SSL Strip (see Section 2.3.1) was analysed in cooperation with the ICS's Network Security Department. As consequence of my security analyses of several web applications run by ICS departments, the discussion about web application security was started at ICS. Today, selected web applications run by ICS departments have implemented techniques increasing their security against SSL Strip attack. There is STS mechanism deployed on the servers together with the client-side scripts alerting users to a potentially unsecured connection.

Besides my doctoral study at the Faculty of informatics, I am employed as a researcher at CESNET. I am interested in development of tools for the security monitoring of high-speed networks. The results of my work are closely connected to the Liberouter project. In the last period I have been focusing on re-adjustment of standard network monitoring and forensics tools for the developed hardware-accelerated devices. This work is described in the article *Hardware Acceleration for Cyber Security* published in proceedings of *IST-091 – Information Assurance and Cyber Defence*. The reprint of the article is available as part of Appendix B. In a connection to my doctoral work I am concerned in possibilities of using network security monitoring and forensics methods currently used in high-speed networks in local networks and smart home system networks.

Currently I supervise two master's theses focused on field of network security applicable to SOHO devices. The first work solves implementation and deployment of the MIPS-based honeypot<sup>1</sup> and the second one is focused on a detection of transparent proxies.

---

<sup>1</sup>A honeypot is a trap set to detect unauthorized use of a system.

## A.1 Publications

### In Conference Proceedings

- Čeleda, P., Krejčí, R., Vykopal, J. and Drašar, M. *Embedded Malware – An Analysis of the Chuck Norris Botnet*. In *European Conference on Computer Network Defense*. Los Alamitos, CA : IEEE Computer Society, 2010. p. 3-10, 8 pp. ISBN 978-1-4244-9377-7.
- Drašar, M., Vykopal, J., Krejčí, R. and Čeleda, P. *Aspect-based Attack Detection in Large-scale Networks*. In *Recent Advances in Intrusion Detection*. Berlin : Springer, 2010. p. 488-489, 2 pp. ISBN 978-3-642-15511-6.
- Novotný, J., Čeleda, P., Dedek, T. and Krejčí, R. *Hardware Acceleration for Cyber Security*. In *IST-091 – Information Assurance and Cyber Defence*. Antalya (TUR) : NATO Research and Technology Organization, 2010. 15 pp. ISBN 978-92-835-0115-2.
- Krmíček, V., Vykopal, J. and Krejčí, R. *Netflow Based System for NAT Detection*. In *Co-Next Student Workshop '09: Proceedings of the 5th international student workshop on Emerging networking experiments and technologies*. New York, NY, USA : ACM, 2009. p. 23-24, 2 pp. ISBN 978-1-60558-751-6.
- Krejčí, R., Lhotka, L., Čeleda, P. and Špringl, P. *Secure Remote Configuration of Network Devices – a Case Study*. In *CESNET Conference 2008 Proceedings*. pages 77-84, 8 p. ISBN 978-80-904173-0-4.

### In Reviewed Journals

- Krejčí, R. and Čeleda, P. *(Ne)bezpečné HTTPS – část II*. Článek v recenzovaném časopise *Data Security Management*. Praha : TATE International, s.r.o., 2010, ročník 14, číslo 4, p. 22–27, 5 pp. ISSN 1211-8737.
- Krejčí, R. and Čeleda, P. *(Ne)bezpečné HTTPS – část I*. Článek v recenzovaném časopise *Data Security Management*. Praha : TATE International, s.r.o., 2010, ročník 14, číslo 3, p. 38–41, 4 pp. ISSN 1211-8737.
- Čeleda, P. and Krejčí, R. *Na stopě Chucka Norrise*. Článek v recenzovaném časopise *Data Security Management*. Praha : TATE International, s.r.o., 2010, ročník 14, číslo 2, p. 30–33, 4 pp. ISSN 1211-8737.

### Others

- Čeleda, P., Krejčí, R., Barienčík, J., Elich, M. and Krmíček, V. *HAMOC – Hardware-Accelerated Monitoring Center*. Technical report. CESNET z.s.p.o., 2010.
- Krejčí, R. and Čeleda, P. *NETCONF – Secure FlowMon Probe Remote Configuration*. In *TERENA Networking Conference 2008*.
- Krejčí, R., Čeleda, P., Špringl, P. and Žižlavský, M. *FlowMon Probe Network Configuration*. Technical report. CESNET z.s.p.o., 2007.
- Čeleda, P., Kováčik, M., Krejčí, R., Kysela, J. and Špringl, P. *Software for NetFlow Monitoring Adapter*. Technical report. CESNET z.s.p.o., 2005.

## A.2 Presentations

- Cooperation on presentation *Hardware Acceleration: An Essential Part of Cyber Security in High-Speed Networks* held in Vienna at *DeepSec – In-Depth Security Conference 2010*.
- Talk at course *PV210 – Security analysis of network traffic* on the theme of *The attack against HTTPS protocol* held in Faculty of Informatics, Masaryk University in Autumn 2010.
- Presentation of my research work and discussion at *Informatic seminar* in Autumn 2009.
- Several presentations for Liberouter team at Liberouter seminars.

## A.3 Posters

- Krejčí, R., Čeleda, P., Krmíček, V. and Novotný, J. *IPFIX Based Virtual Network Monitoring*. *TERENA Networking Conference 2009*.
- Krmíček, V., Čeleda, P. and Krejčí, R. *Hardware-Accelerated Framework for Flow Monitoring of 10Gbit Networks*. *TERENA Networking Conference 2008*.

## A.4 Participation in Projects

- Senior software developer in research team of the Liberouter project.

## A.5 Teaching

In the Autumn 2010 I taught course *PB173 – Domain specific development in C/C++* focused on Linux system programming.

## A.6 Passed Courses

- IA067 Informatics Colloquium
- IA068 Seminar on Informatics
- IA158 Real Time Systems
- PA163 Constraint programming
- PA174 Design of Digital Systems II
- PA176 Architecture of Digital Systems II
- PB170 Seminar on Digital System Design
- PV193 Accelerated Algorithms
- PV194 External Environments of Digital Systems

## A. SUMMARY OF MY PRESENT WORK

---

- PV205 Seminar on Complex systems
- PV210 Security analysis of network traffic
- VV041 English for Academic Purposes

## Appendix B

### Reprints of Selected Papers

- Čeleda, P., Krejčí, R., Vykopal, J. and Drašar, M. *Embedded Malware – An Analysis of the Chuck Norris Botnet*. In *European Conference on Computer Network Defense*. 1. ed. Los Alamitos, CA : IEEE Computer Society, 2010. p. 3-10, 8 pp. ISBN 978-1-4244-9377-7.
- Novotný, J., Čeleda, P., Dedek, T. and Krejčí, R. *Hardware Acceleration for Cyber Security*. In *IST-091 – Information Assurance and Cyber Defence*. Antalya (TUR) : NATO Research and Technology Organization, 2010. 15 pp. ISBN 978-92-835-0115-2.