The Amplification Threat Posed by Publicly Reachable BACnet Devices

Oliver Gasser[1], Quirin Scheitle[1], Benedikt Rudolph[2], Carl Denis[1], Nadja Schricker[1] and Georg Carle[1]

[1]*Technical University of Munich, Germany*
[2]*DE-CIX, Germany*
Email: {gasser, scheitle, denis, schrickn, carle}@net.in.tum.de; benedikt.rudolph@de-cix.net

Received 20 September 2017; Accepted 24 September 2017;
Publication 20 October 2017

Abstract

In a connected world Internet security is becoming increasingly important. Attacks, which are frequently executed by botnets, can impact people in their everyday life. A ubiquitous kind of attack is the amplification attack, a special type of Denial-of-Service attack. Several protocols such as DNS, NTP, and SNMP are known to be vulnerable to amplification attacks when security practices are not followed. In this work we evaluate the vulnerability of BACnet, a building automation and control protocol, to amplification attacks. To assess BACnet's vulnerability we conduct active traffic measurements on an Internet-wide scale. We find 16 485 BACnet devices, the largest number to date. Additionally, more than 14 k of these devices can be misused as amplifiers, with some generating amplification factors up to 120. To remediate this potential threat we employ a vulnerability notification campaign in close coordination with a CERT. We assess the success of the campaign and find that the number of publicly reachable BACnet devices decreased only slightly. Additionally, we employ passive measurements to attribute the majority of BACnet traffic in the wild to scanning projects. Finally, we also give suggestions to thwart the amplification attack potential of BACnet.

Journal of Cyber Security, Vol. 6_1, 77–104.
doi: 10.13052/jcsm2245-1439.614

Keywords: BACnet, building automation, network scan, amplification attack, notification.

1 Introduction

In the last years the number of Denial-of-Service attacks increased dramatically in both frequency and data rate. These attacks more and more misuse Internet of Things (IoT) devices or embedded systems. Many of these devices are not properly secured and are therefore an ideal target for misuse and attacks. They can be used directly by being part of a botnet, or indirectly as a reflector or amplifier. An example for direct abuse is the Mirai botnet which attacked the Internet infrastructure company Dyn causing partial outages for Twitter, Amazon, and Netflix [16], and started a DDoS attack which Akamai was unable to mitigate [17]. An example of indirect abuse is the use of open DNS resolvers as amplifiers in the attack on Spamhaus [23].

These examples highlight problems arising from two sources: Firstly, IoT devices without proper security posture may be taken over for arbitrary abuse. Secondly, embedded devices may offer insecure and easy to abuse services such as open DNS resolvers and misconfigured NTP servers. Most of these security problems, however, are only discovered when their exploitation causes fallout. It is therefore crucial to identify potentially insecure devices before they are being misused in attacks. We focus our measurements on the building automation protocol BACnet [2] and assess its vulnerability to amplification attacks. BACnet is capable of connecting a wide range of devices and offers remote monitoring and control features. Security was not a priority when the BACnet protocol was designed and the recommendation [20] is to never connect BACnet devices to the Internet, but always place them in a segmented, separate network. We investigate whether this recommendation is followed by probing for BACnet devices which are reachable in the public Internet.

Our contributions are as follows:

- Conducting exhaustive, Internet-wide scans for BACnet devices, varying port and payload
- Discovering the largest number of BACnet devices to date
- Uncovering and quantifying the potential of BACnet for amplification attacks
- Evaluating passive traffic data to understand BACnet traffic in the wild
- Executing a CERT-backed notification campaign and evaluating its success using active and passive measurements
- Recommending specific security improvement steps

Outline: This paper is structured as follows: In Section 2 we briefly describe the BACnet protocol and our choice of scanning payload. We continue with our scanning methodology and ethical considerations in Section 3. Section 4 details the BACnet deployment evaluation based on our scan results. In Section 5 we analyze in detail how BACnet devices can be used for amplification attacks. Section 6 investigates BACnet traffic seen in the wild using passive traffic measurements. In Section 7 we detail our CERT-backed notification campaign and assess its success. Additional efforts to remediate the threat posed by publicly accessible BACnet devices are discussed in Section 8 and related work is presented in Section 9. Section 10 concludes this paper with a summary and an outlook for future work.

2 The BACnet Protocol

This section provides a brief overview of the BACnet protocol, highlighting aspects important for this research.

BACnet development was started in 1987 [20], with the first release in 1995 by ASHRAE. BACnet was designed as a standalone network protocol, including its own network layer with 16-bit network and device identifiers. BACnet's dedicated network layer implied segmented networks, hence security was not a consideration in protocol design. In 1999, BACnet/IP was defined to use IP as the network layer, which comes with many security implications. Security advice for BACnet/IP to date is to segment BACnet networks.

BACnet/IP uses a rather complex packet structure with multiple internal header layers. In its design, BACnet properties somewhat resemble SNMP MIBs.

2.1 BACnet Payload

For our measurements we use the generic wild-card device ID `0x3fffff` and select the following suitable payloads to identify BACnet devices:

IPv4: We conduct IPv4 measurements using a *ReadPropertyMultiple* request payload. This type of request allows to specify a list of BACnet property IDs (*e.g.,* `0x46` = model name, `0x79` = vendor name). The queried BACnet device returns a list of corresponding property values (*e.g.,* model name: Niagara AX, vendor name: Tridium Inc.).

IPv6: IPv6 support for BACnet was added in 2016 [3]. The standard defines new header types for IPv6, requiring a different payload to identify

IPv6-capable devices. We use a *VirtualAddressResolution* request to discover BACnet devices over IPv6. IPv6-capable BACnet devices return the remote virtual address which is needed in the subsequent *ReadPropertyMultiple* request.

Amplification: The payload in our amplification scans is amended with additional properties which promise a high amplification factor, such as *PropertyList*.

3 Methodology

This section describes our methodology by giving details on our active scans, the processing of answers, and ethical considerations guiding our research.

3.1 Scan Overview

BACnet is run on UDP ports 47808 – 47823 by default [2]. Using different strategies, we probe those ports via IPv4 and IPv6. We verify responses for valid payloads to filter for actual BACnet devices. Using a different scanning payload, we then further survey these BACnet devices to determine their vulnerability for amplification attacks. Depending on the number of targets, we optimize packet sending rate to (1) minimize network load and (2) achieve tractable scanning duration. Table 1 gives an overview of the scan types, listing the number of conducted scans, the number of scanned ports, the used packet rate, the scan duration, the number of targets, received responses ("Resp."), and parsable BACnet payloads ("BACnet").

3.2 Internet-wide IPv4 Scans

We probe the IPv4 address space on the previously mentioned UDP ports using ZMap [10] and a BACnet UDP payload. We exclude IP addresses that

Table 1 Overview of all BACnet scans

Type of Scan	Number of Scans	Ports	Rate	Duration	Targets	Resp.	BACnet
IPv4-wide	4	16	25 kpps	41 h	2.4 G	32 868	16 485
IPv6 hitlist	1	1	5 kpps	2 min	407 k	0	0
Amplification	1	16	100 pps	3 min	16 k	15 598	15 429

are (1) on our blacklist or (2) part of the IANA reserved ranges [14] or (3) not routed according to BGP data from our routers.

For the IPv4 scans we choose a rate of $25\,\mathrm{kpps}$, resulting in a duration of 41 hours for each performed scan. The scans are run from four measurement machines located in a dedicated measurement network.

We conduct four IPv4-wide scans: The first scan was executed in December 2016 and is used to evaluate the BACnet deployment (see Section 4). The three subsequent IPv4-wide scans are conducted in February, July, and August 2017. We use those to assess the success of the notification campaign (see Section 7).

We use the same filtering process to identify valid responses for all scans. In the following we describe this process and show breakdown numbers from the December 2016 scan.

In a first step, we filter the raw ZMap results for packets with the queried source port. We discard about $20\,\%$ of mismatching responses, which stem from source ports such as UDP/53 (DNS) or UDP/39999 (unregistered, but linked to Sygate [7]). These responses might be counter-scans from infected or malicious devices, probing our IP address for vulnerabilities. After this filtering, we count responses from $32\,\mathrm{k}$ unique IP addresses. The scan on port 47808 produces about $17\,\mathrm{k}$ ($53\,\%$) responses, port 47809 about $3\,\mathrm{k}$ ($9\,\%$), port 47810 about $1.1\,\mathrm{k}$ ($3\,\%$), and ports $47811-47823$ hold about equal shares of the remaining $35\,\%$. This result supports our decision to scan for all 16 official BACnet ports to obtain a complete picture of the BACnet deployment. Scanning only the most prominent port UDP/47808 as *e.g.,* done by Mirian *et al.* [19] misses about $47\,\%$ of publicly reachable BACnet IP-port combinations.

In a second step, we filter the responses for valid BACnet payloads. We use our tailor-made Python BACnet module which we publish on GitHub [12]. We filter for compliance with the following characteristics, which are required for a genuine response to our packet: The transport type is BACnet/IP (`0x81`), the payload is an original unicast NPDU (`0x0a`), the BACnet version is the only valid version 1 (`0x01`), no reserved NPDU control bit is set, and the application payload type is BACnet-ComplexACK-PDU (`0x03`). After the filtering phase $16\,485$ (of initially $32\,868$) valid BACnet responses with payload content remain. Spot-checks on non-compliant packets reveal payloads that are *e.g.,* invalid, mirrored, randomized, or associated to other protocols. These might stem from honeypots, BACnet simulators, or unusual device configurations. Further investigation of these devices would require more intrusive scanning.

By scanning all standardized BACnet ports we also obtain more valid BACnet payloads: Our $16.4\,\mathrm{k}$ valid responses exceed Mirian *et al.*'s $12.8\,\mathrm{k}$ "valid handshakes" [19].

The distribution of ports after this filtering is more centric towards port UDP/47808 ($84.4\,\%$ of responses). We evaluate the responses from all four scans in detail in Sections 4 and 7.

3.3 IPv6 Scans

As IPv6 support for BACnet was added in early 2016 [3], we scan for BACnet devices in the IPv6 space.

Since scanning the full address space is not feasible in IPv6, we follow the domain-resolution approach of our IPv6 hitlist [13]. We also gain IPv6 addresses from responsive IPv4 BACnet devices by querying their rDNS record for `AAAA` records. We query $407\,\mathrm{k}$ unique IPv6 addresses, but do not receive any reply. We argue that this is likely due to a lack of IPv6 support in the field. As BACnet simulators do not support IPv6 yet, we can not validate our payload, which we thoroughly check against the BACnet standard.

3.4 Amplification Scans

Based on the subset of responsive BACnet devices, we conduct additional scans to evaluate the amplification potential of those devices. Compared to previous scans we now request additional BACnet properties. Since these scans might produce more load on target systems we reduce the scanning rate to 100 packets per second. We apply the same filtering steps as for the IPv4-wide scans. This removes about 170 responses from non-scanned IP addresses.

3.5 Ethical Considerations

We follow an internal multi-party approval process before any measurement activities are carried out. This approval process incorporates the proposals of Partridge and Allman [21] as well as Dittrich *et al.* [8]. We assess whether our measurements can induce harm on individuals in different stakeholder groups. As we use a valid payload in accordance to the BACnet standard, it is unlikely for our scans to cause problems on scanned devices. We minimize interference of our scans by following best scanning practices such as maintaining a blacklist and using dedicated servers with informing rDNS names,

web sites, and abuse contacts. We consider that publication of IP addresses of possibly vulnerable and amplifying devices may be abused by third parties. The conclusion of this process is that it is ethical to conduct the experiment, but that we will, in contrast to our usual policy, not share data from this work with the public. Instead, we will only make the data available upon request to other researchers for reproducibility and comparison, and to the DFN-CERT for vulnerability notification of affected parties. During our scans we did not receive any complaints.

4 BACnet Deployment

In this section we evaluate the BACnet deployment by analyzing the responses obtained from our December 2016 scans.

4.1 Vendor Analysis

We find devices from a total of 97 different vendors, with just the top 3 vendors representing 52 % of all devices. Table 2 shows the five most frequent vendors found in our scans. Mirian *et al.* [19] also find Reliable Controls (12.7 %) and Tridium (10.6 %) as their top BACnet vendors, however the share of these vendors in our evaluation is larger.

4.2 Topological Clustering

We next investigate the distribution of BACnet devices over Autonomous Systems (ASes) and announced prefixes. We use CAIDA's routeviews data [5] to map IP addresses.

We find AS coverage rather sparse, with BACnet devices present in 1439 ASes, with a median of 2 devices per AS. This is a small share of the 55 738 total ASes [5].

Table 2 Top 5 BACnet vendors in results

Pos.	Vendor ID	Vendor Name	Count	%
1	35	Reliable Controls Corporation	3740	24.8
2	36	Tridium Inc.	2079	13.8
3	8	Delta Controls	2004	13.3
4	5	Johnson Controls Inc.	1328	8.8
5	24	Automated Logic Corporation	1051	7.0

Table 3 Top 5 ASes by count of BACnet devices

Pos.	ASN	Organization	Count	%
1	7018	AT&T Services, Inc.	1510	9.2
2	7922	Comcast Cable Communications, Inc.	1450	8.8
3	22394	Cellco Partnership DBA Verizon Wireless	774	4.7
4	852	TELUS Communications Inc.	697	4.3
5	6327	Shaw Communications Inc.	454	2.8

We also find our number of 1439 ASes to be in line with Mirian *et al.*, who discover BACnet devices in 1330 ASes.

The BACnet devices from our scans cover 5109 announced prefixes, of which 3021 only contain 1 device. The top 5 prefixes are /16 or larger prefixes of the major Internet service providers highlighted in Table 3.

4.3 Geographical Clustering

We also map the IP addresses of BACnet devices to countries using the IP2Location database [1]. While research has shown that IP geolocation databases can introduce significant biases [22], we believe them still to be indicative of the top countries of deployment. We find BACnet devices to be very centrally clustered with 60 % in the US and 20 % in Canada. With significantly less devices, Australia (3 %), France (2 %) and Spain (2 %) follow.

5 Amplification Attacks using BACnet

This section describes BACnet's vulnerability to amplification attacks. We evaluate the number of available amplifiers as well as the bandwidth amplification factor (BAF) of BACnet. BACnet supports both single property and multi property requests. To assess the amplification potential, we scan with a generic multiple property payload. From this, we derive (1) empirical BAF for our generic payload, (2) calculated BAF for individual properties in a single property request, and (3) calculated BAF for individual properties when repeatedly requesting the specific property in a multiple property request.

5.1 Amplification Attack Characteristics

An amplification attack is a type of Denial-of-Service attack where (1) the response payload is larger than the request payload. This ratio is called the

bandwidth amplification factor (BAF) [25]. In addition to a BAF >1, there are two other typical characteristics for amplification attacks: (2) the used protocol is stateless and (3) no authentication is required.

BACnet/IP is a UDP-based protocol and does not require any handshake. This stateless property already satisfies characteristics (2) and (3). Since we are free to choose the requested property which the BACnet device will then answer (provided the device supports the property), we can select properties which will most likely trigger a large response by the queried device. In the following we evaluate which properties provide us with a large BAF. If such a property is found, characteristic (1) is satisfied and BACnet can be used in amplification attacks.

5.2 Number of BACnet Amplifiers

We find $15\,429$ responsive BACnet devices with our amplification scans on ports 47808 – 47823. If a device does not support a requested property, it will reply with a four byte error, resulting in a property BAF < 1. We quantify the amplification attack threat per BACnet property and device using error-free responses only. We focus the amplification attack analysis on variable length properties (*i.e.,* strings or arrays) as these are more likely to give a larger BAF.

In Table 4 we see stark differences in the number of available amplifiers depending on the requested BACnet property: Most properties provide us with about $14\,\mathrm{k}$ amplifiers, whereas three properties are available on significantly fewer devices: 2316 ($15.0\,\%$) of BACnet devices tell us their serial number, 1958 ($12.7\,\%$) give information about their profile name, and 1389 ($9.0\,\%$) provide their list of available properties. We investigated the reason for this and found that many devices answered with the BACnet error *property unknown* for these three properties. This is not surprising as the properties *serial number*, *profile name*, and *property list* were only added in 2012 to the BACnet standard. In conclusion, this analysis shows that we need to take the different numbers of amplifiers into account when trying to assess the potential threat posed by BACnet-based amplification attacks.

5.3 Amplification Factor of Scanning Payload

Figure 1 shows the empirical CDF of the bandwidth amplification factor for our 49 bytes long scanning payload. We can see that more than $90\,\%$ of requests generate responses with a BAF ≥ 5. The median BAF is 9, and the maximum BAF is 19.8 (with a response payload length of 942 bytes).

Table 4 Property BAF and payload BAF as mean over *all*, top *50 %* and top *10 %* amplifiers

		Property BAF			Payload BAF		
Property	Amplifiers	All	50 %	10 %	All	50 %	10 %
model_name	14 072	6.2	8.3	8.5	1.5	1.7	1.7
vendor_name	14 072	9.0	13.9	14.5	1.8	2.2	2.3
firmware_revision	14 072	11.2	19.6	35.0	2.0	2.8	4.2
app_sw_version	14 071	5.9	10.3	14.0	1.5	1.9	2.2
object_name	14 039	6.8	9.1	11.0	1.6	1.8	2.0
description	13 741	5.5	10.9	13.0	1.4	1.9	2.1
location	13 360	2.5	5.1	7.5	1.1	1.4	1.6
serial_number	2316	4.9	5.6	5.0	1.4	1.4	1.4
profile_name	1958	5.0	7.0	7.0	1.5	1.8	1.8
property_list	1389	141.0	193.8	200.0	7.3	9.7	10.0

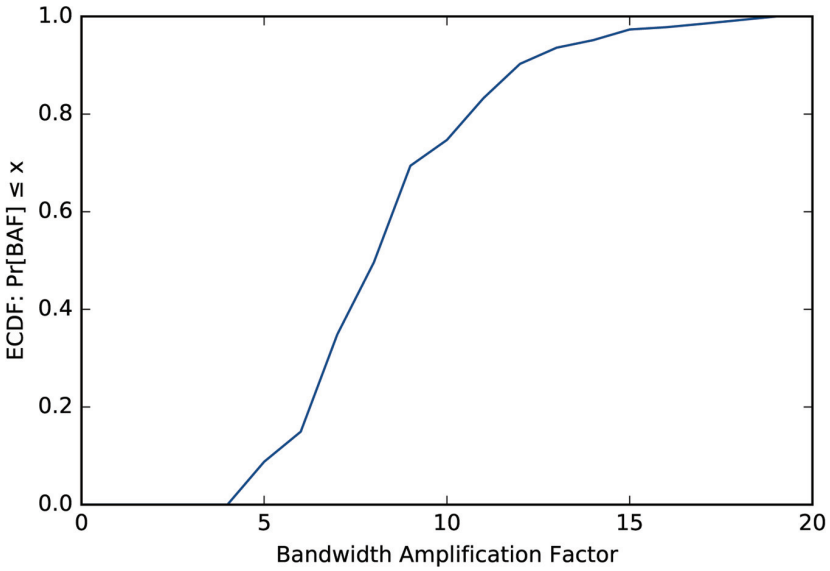


Figure 1 Distribution of BAF for our generic *ReadPropertyMultiple* amplification payload used in scans.

5.4 Amplification Factor per Property

We now evaluate the BAF on a per property basis *i.e.,* if we would send a request for a single property. To this end, we first calculate the sending and receiving overhead of BACnet headers and the static part of the payload. The sending ($SEND\_OVERHEAD$) and receiving

($RECV\_OVERHEAD$) overhead caused by BVLC, NPDU, and APDU headers in addition to the static part of the BACnet payload is 19 bytes. When requesting a property we need to add 2 or 3 additional bytes to the sent payload, depending on the property ID ($prop\_id\_len$). With the response property length ($prop\_len$), we can now calculate the BAF for a single property payload as follows:

$$BAF = \frac{RECV\_OVERHEAD + prop\_len}{SEND\_OVERHEAD + prop\_id\_len}$$

Table 4 shows a per-property BAF analysis: Property BAF details the length ratio of returned property and queried property ID. Payload BAF shows the received and sent payload length ratio for a packet requesting only this property.

We can see that *property list* has by far the largest property BAF with an average of 141. On the other hand, more than ten times as many amplifiers are available for properties such as *description*, *location* or *model name*. The property *firmware revision* combines many available amplifiers with a high BAF.

Due to the overhead introduced by BACnet headers, the payload BAF is much smaller than the property BAF.

5.5 Tuning the BACnet Payload

When issuing a request for a single property (as simulated with payload BAF in Table 4), the amplification potential of BACnet is not fully leveraged. Requesting multiple properties in the scanning payload can significantly increase the payload BAF. Figure 1 shows that our multi-property scans generate a median payload BAF of 9, exceeding all single-property mean payload BAFs in Table 4.

To raise the payload BAF even further, we can tailor a payload of multiple requests of the same property with a high property BAF factor. We very carefully test this behavior with a small number of BACnet devices. The devices not only answer the request without error, but also send the property multiple times. This allows us to leverage the property BAF, minimize the overhead of BACnet headers, and hence boost payload BAF factors up to 120.

Figure 2 shows the distribution of payload BAF when the same property is requested multiple times. In this comparison, we choose the properties *property list* as it provides the largest average BAF and *firmware revision*

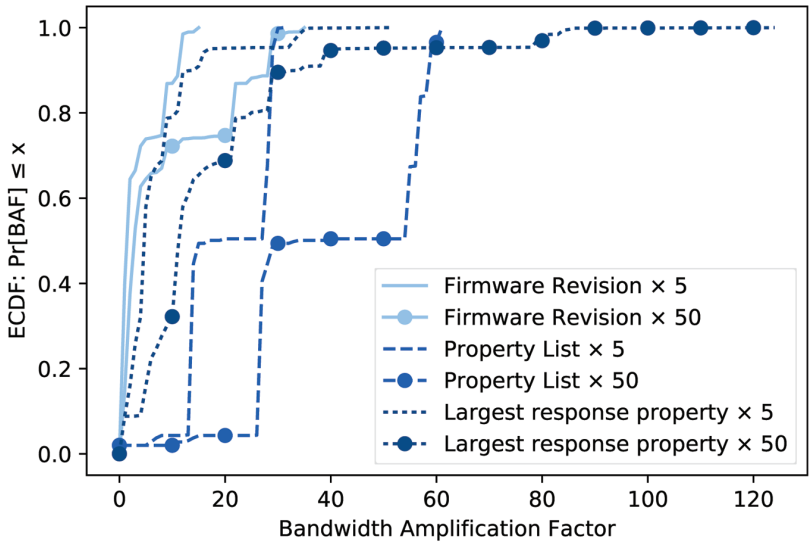


Figure 2 Payload BAF when issuing multiple requests for the same property (within a single Multi-Property packet).

as it has the most amplifiers with second largest average BAF. Additionally, we include the property triggering the largest response on a per-device level. The influence of BACnet headers decreases when we increase the number of requested properties from 5 to 50.

The majority of the amplifiers answering *firmware revision* requests give us a BAF below 10. About 10 % offer a BAF of about 30 when requesting this property 50 times.

Requesting *property list* five times already generates a larger BAF than 50 requests for *firmware revision*. About half of the 1389 amplifiers generate a BAF of 27 and 55, for 5 and 50 requested properties respectively. This BAF is larger than for SNMP-based amplification attacks and similar to those exploiting open DNS resolvers [25].

The distinctively noticeable steps in Figure 2's *property list* distributions are a result of vendor clustering: Devices produced by Trane, which occur 449 times, always have a *property list* length of 93 bytes. We found that all devices by Reliable Controls send a 188 bytes or longer *property list*. This is a consequence of the large number of properties supported by these devices. However, it also means that these devices are particularly valuable targets for attackers who want to misuse them in amplification attacks.

Using the largest property on a per-device level includes all BACnet devices and gives us a higher BAF than *firmware revision*. $30\,\%$ of all BACnet devices allow for a BAF of 20 or larger. This type of attack, however, is more complex than simply choosing a single property: A preceding reconnaissance scan to find the largest property for each device and a device-specific payload would be necessary.

6 BACnet Traffic in the Wild

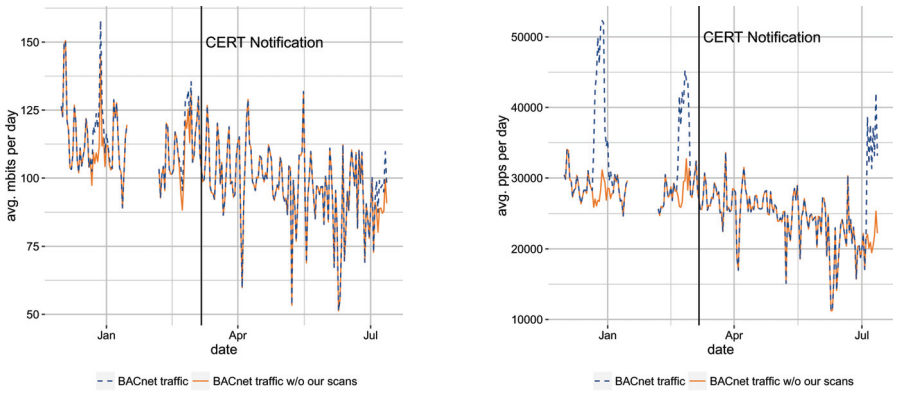
In this section we evaluate BACnet traffic as observed in the wild through two vantage points: First, we look at flow data of a large European IXP. Second, we analyze raw packet data at a Japanese research backbone network.

6.1 IXP Flow Data

Our first vantage point at a large European IXP allows us to obtain an authentic view of BACnet traffic in the Internet [6]. The IXP is located in central Europe and interconnects about 700 ASes which exchange more than $5\,\mathrm{Tbit/s}$ at peak times. We rely on flow data from the IXP's switching fabric, where we sample every 10,000th packet from December 1, 2016 until July 12, 2017. Due to technical issues flow data is missing between January 16 and February 5.

To preserve comparability with active scanning we filter UDP traffic on all 16 BACnet ports. We remove traffic with ports < 1024 as these are very likely cases where a BACnet port was randomly chosen as an ephemeral client port. We also identify traffic from our own active BACnet scans by source IP address. Figure 3a depicts the BACnet traffic volumes in Mbit/s at the IXP for the measured period (99.66% IPv4, 0.34% IPv6). We notice a spiky pattern that indicates frequent scanning activities (max 150, min 51 Mbit/s) and we can indeed clearly observe our active scans. The traffic levels decline continuously to less than 100 Mbit/s after DFN-CERT issued the advisory (CERT notification). If we consider packets per second, see Figure 3b, for the same vantage point and duration we confirm our findings: the number of packets decreases after the CERT notification and the scan patterns are visible. In fact, our own scans are even more visible, i.e., the peaks are relatively higher.

Next, we analyze the distribution of transport layer ports for BACnet traffic seen at the IXP. The distribution of used ports is quite different for source and destination, as can be seen in Table 5. The source port distribution



(a) BACnet traffic volume at the IXP.

(b) BACnet packets per day at the IXP.

Figure 3 BACnet traffic at the IXP.

Table 5 Top 5 source and destination ports of BACnet traffic in IXP dataset, ordered by destination port

Port	Dst %	Src %
47808	21.34	2.82
47820	2.71	2.83
47822	2.66	2.84
47816	2.64	2.71
47810	2.61	2.56

is dominated by BACnet ports, evenly distributed from 2.82% (47808) to 2.24% for 47819. The top non-BACnet source port is 7985 (1.83%). The distribution of destination ports is different. Port 47808 is the most frequent one and accounts for 21.34% of all seen BACnet flows. The other BACnet ports are again evenly distributed and in the range from 2.7 to 2.2%. In both distributions we find (after the BACnet ports) a small fraction of UDP application ports, each with a share of $< 1\%$ (except source port 7985). This is most likely UDP application traffic on non-privileged ports where the client has accidentally chosen a BACnet port as an ephemeral port. The port distribution indicates that the majority of BACnet traffic stems from scans on port 47808. Scanners use 47808 as their destination port, but get few responses which is why the percentage of 47808 on the source port side is much lower.

In Figure 4, we plot the distribution of the packet size in bytes of the BACnet traffic. While small packets between 66 and 83 bytes are the most

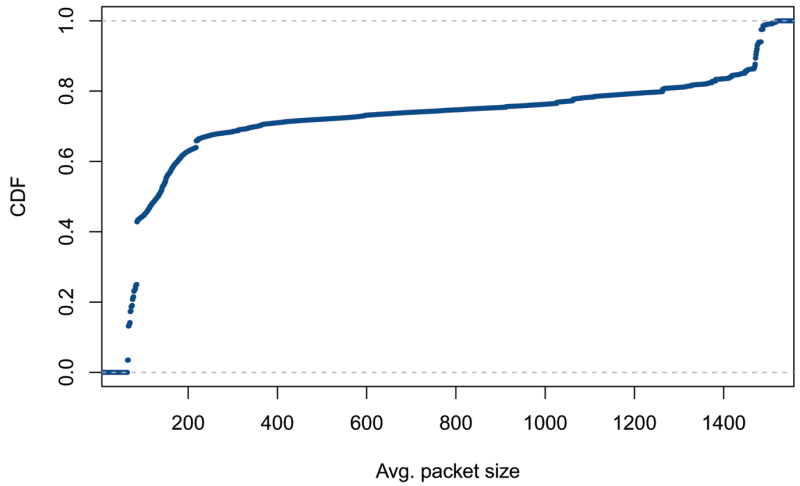


Figure 4 CDF of the average BACnet traffic packet size.

prevalent, there is also a significant number of packets larger than 1400 bytes. This hints at a large number of small sized scanning packets (typically around 45 to 70 bytes) in combination with application layer UDP data where a BACnet port is chosen as the ephemeral port.

In summary, a large portion of BACnet traffic seen at the IXP is most likely scanning traffic. We analyze this phenomenon more in depth using raw packet data in the following section.

6.2 MAWI Raw Packet Data

To gain additional insight into BACnet traffic beyond flow data, we analyze raw packet data from our second vantage point, the MAWI data set [28]. We use 48 hour traces captured at the transit link of the WIDE research network to the upstream ISP on April 12–13, 2017. The traces comprise of 3.9 G packets with more than 12 TB of data, resulting in an average data rate of 550 Mbit/s.

We first filter the data set to UDP traffic on all 16 BACnet ports. 403 837 packets are remaining after this step. We then remove traffic where a BACnet port was chosen as an ephemeral client port. We do this conservatively by eliminating traffic where we find a low port ($<$1024). By looking at the payload of top non-BACnet ports we find additional occurrences of BACnet as an ephemeral port and remove Teredo, SSDP, and Netis router backdoor scans. After this port filtering stage 339 274 packets with traffic on BACnet ports remain.

Next, we parse the UDP payload to find out whether this traffic is in fact BACnet traffic. We identify BACnet/IP traffic by filtering for the the distinct BACnet/IP transport type (`0x81`) at the beginning of the UDP payload. We also try to identify BACnet/IPv6 traffic (transport type `0x82`), but could not find any. Interestingly, however, we find 31 packets with BACnet/IP payloads built for IPv4 but sent over IPv6. We also check the valid BACnet version (`0x01`) and ensure that no reserved NPDU control bit is set. In contrast to our filtering procedure for active scans, we do not restrict the payload to BACnet-ComplexACK-PDU, but allow all valid BACnet application payload types. In the payload filtering phase we remove 4120 packets, with $335\,154$ packets with BACnet payload remaining.

We analyze the BACnet payload of the remaining packets which are all destined to port UDP/47808. Surprisingly, these $335\,154$ packets contain only four different payloads. All four payloads are BACnet requests, no responses are present. This hints at scanning activities instead of regular BACnet traffic.

Table 6 shows an overview of the four payloads, with the number of seen packets, number of source IP addresses, requested BACnet properties, and a classification of the scan.

The most common payload #1 contains a Multi-Property request querying a list of 10 properties. It is sourced from 10 IPv4 addresses in different subnets and Autonomous Systems. The reverse DNS name mapping of two of the IP addresses hints at rented private servers, one rDNS entry hints at a research project (*thisissecurityresearch.com*). We find a web server running on eight of the ten IP addresses, informing of a "Short Time Scanning Project" and giving the possibility to be excluded from scans.

Payload #2 requests the object ID and occurs in more than $66\,\mathrm{k}$ packets. We find 25 IPv4 source addresses and one IPv6 source address. The single IPv6 address, however, sends a BACnet/IP payload instead of a correct BACnet/IPv6 payload. We attribute 23 of the 25 IPv4 addresses to the scanning service Shodan [27] due to the reverse domain name mapping.

Table 6 BACnet packets in MAWI dataset classified according to their payload

#	Packets	Source IPs	Req. Properties	Classification
1	263 273	10	List of 10 properties	Short Time Scanning Project
2	66 670	26	Object ID	Shodan
3	4441	1	Vendor ID	Chinanet
4	770	242	List of 9 properties	Kudelski Security

The third most common payload #3 requests the vendor ID and stems from only a single IPv4 address without an rDNS entry. The IP address belongs to the Autonomous System of Chinanet, a Chinese ISP.

The least common payload #4 consists of a Multi-Property request containing 9 properties. Its 242 IPv4 addresses are all located within the same /24 subnet belonging to Kudelski Security, a company offering Internet security services. The WHOIS entry also states that the network is used for port scanning activities.

Next, we analyze the temporal scanning patterns of the four payloads. Figure 5 shows the number of packets seen per hour for each different payload. We can clearly see distinct scanning patterns: The "Short Time Scanning Project" conducts high-rate scans with more than 40 k packets per hour. These high-rate scans, however, only last for some hours, after which they decrease in rate and vanish completely. This bursty phenomenon could be due to non-random scanning, where adjacent IP addresses are probed close after each other. Shodan on the other hand continuously scans for BACnet devices over the two day period with a very constant packet rate. In the MAWI dataset we find about 1000 packets each hour originating from Shodan IP addresses. Chinanet scans are only observed in the first six hour period, exhibiting a relatively constant packet rate. Kudelski Security seems to be conducting brief daily scans, which we observed at the same hour of the day.

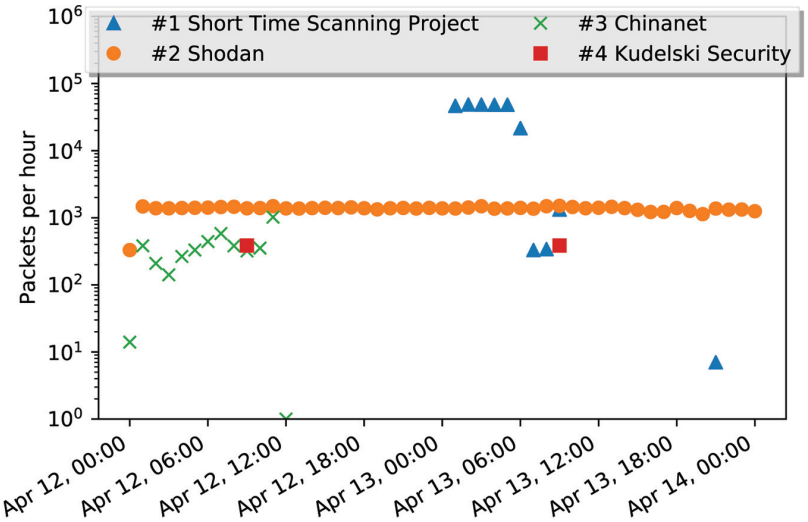


Figure 5 Packets per hour for each of the four payloads. Continuous vs. burst scanning clearly visible. Note that the y-axis is log-scaled.

To summarize, the MAWI dataset gives us a glimpse into BACnet traffic, specifically scanning practices. We do not find any bidirectional BACnet traffic as we see only BACnet requests. This hints at port scans of which the majority seems to be conducted by security companies and researchers. We identify clear temporal scanning patterns based on the four payloads.

7 Notification Campaign

We use our measurement results to improve Internet security by notifying the owners of affected BACnet devices. We cooperate with the DFN-CERT, which is the Computer Emergency Response Team for the German National Research and Education Network (DFN). We supply the DFN-CERT with relevant information of the affected systems. The DFN-CERT notified 76 different CERT teams and additionally the CERT Coordination Center for affected systems in the US and Canada. The notifications were sent out in the week of March 7, 2017. By notifying vulnerable systems we hope to reduce the number of publicly reachable and abusable BACnet devices. Li *et al.* show that notification campaigns can drive measurable impact [18]. We assess the impact of our notification campaign using follow-up active scans and passive analysis at a large European IXP.

7.1 Follow-up Active Scans

To assess the impact of our notification campaign we conduct four IPv4-wide BACnet scans. Table 7 shows all four scans: The first was conducted in December 2016 for BACnet deployment analyses. In February 2017 a second scan was performed to get an updated list of IP addresses. This list was given to the DFN-CERT, which conducted the notification in the beginning of March 2017. The third and fourth scans were conducted to assess the impact of the notification campaign, in July and August 2017 respectively.

In Table 7 we see that we receive many more responses in the third and fourth scan. The vast majority of these responses, however, are not genuine

Table 7 IPv4-wide BACnet scans to assess notification campaign impact

Scan Date	Responses	BACnet	Unique IPs	Prefixes	ASes	Unique IDs
Dec 2016	41 103	16 485	15 350	5110	1439	9319
Feb 2017	39 581	16 645	15 495	5159	1465	9392
Jul 2017	758 611	16 351	15 152	5020	1425	9269
Aug 2017	141 567	16 247	15 030	5040	1428	9188

BACnet responses. Instead they are mirrored BACnet request packets containing the exact same payload as our scans. These mirrored packets could be caused by misbehaving or misconfigured routers on the path. As we only evaluate genuine BACnet responses, we remove all mirrored packets. The number of valid BACnet responses and unique IP addresses remains mostly constant over the four scans, with small reductions in the July and August 2017 scans. The number of network prefixes and Autonomous Systems (ASes) with BACnet devices decreases slightly between the second and third scan.

To better understand these slight trends we evaluate the sets of IP addresses in the scans in more detail and correlate them with each other. During this analysis we see that BACnet IP addresses are not steadily going offline as suggested in Table 7, but rather quite unstable. We find $10\,841$ IP addresses which respond to at least one but not all our probes. In each scan about half of these IP addresses are responsive. These IP addresses can be deemed unstable or dynamic.

We compare the AS distribution of the unstable IP addresses to those of all IP addresses but can not find any major differences.

To further analyze the changes between the scans we try to fingerprint BACnet devices. We use the device properties model name, location, object name, vendor ID, and vendor name to create a device ID. We then check whether this ID is unique in each of the four scans. We find about $9\,\mathrm{k}$ unique IDs per scan. Next, we check if the IDs of IP addresses stay the same on subsequent scans. About 400 IP addresses change their unique device ID between subsequent scans. By inspection we deduce that most of these changes are caused by IP address reassignment and the resulting device ID swapping. Additionally, we identify unique device IDs which change their IP address between scans. There are between 220 and 450 of these devices. When inspecting the corresponding AS, between 68% and 87% of these IP addresses belong to the same Autonomous System.

Consequently, we conclude from the follow-up active scans that the majority of dynamic BACnet devices are a result of IP address changes within the same organization. Even though the overall number of publicly reachable BACnet devices has gone down slightly, the notification campaign seems to have had only a small impact.

7.2 IXP Temporal Comparison

In addition to our follow-up active scans we also conduct passive analysis using an IXP data set, presented in Section 6.1. When looking at Figure 3a

and Section 6.1 we see a slight downward trend of BACnet traffic and packets respectively. This again suggests that the notification campaign contributed to an improvement of the situation.

8 Discussion

We use this section to discuss the implications of our results and how to improve the security state of BACnet devices. Accordingly, this section explores: (1) strategies for affected parties to detect and prevent BACnet-based attacks and (2) action that the community can take to remedy the problem of publicly accessible BACnet devices.

8.1 Mitigation Strategies

Affected network operators can adopt various strategies to reduce the impact of BACnet attacks. First, and preferably, the operator of the device can move the device to a separate, not publicly accessible network enforced by, *e.g.,* VLAN or VPN. However, this may not be feasible in many small network scenarios. Second, the network operator may deploy rule-based access filtering to restrict access from the public Internet. Third, at network operator or ISP level, strategies may be deployed to detect ongoing amplification attacks, e.g. by measuring traffic entropy [4]. Detected attacks could be rate-limited by an ISP.

8.2 Standardization Efforts

BACnet standardization could harvest quick wins in mitigating amplification attack potential by not allowing multiple reads of the same property within the same packet, which we found critical in achieving a high BAF. However, this comes with a certain complexity and computational cost. We contacted ASHRAE regarding changing the BACnet standard to thwart the attack potential, but did not receive a reply.

9 Related Work

In this section we elaborate on existing related work in the areas of Internet scanning for BACnet and similar protocols, amplification attacks and vulnerability notification.

9.1 Internet-wide Scanning

Both Mirian *et al.* [19] and Feng *et al.* [11] scan for BACnet and other ICS devices. In contrast to them we scan for all 16 standardized BACnet ports. We identify twice as many IP addresses that do not respond on port UDP/47808 for a total of $3.7\,\mathrm{k}$ valid BACnet responses missed by previous research. Neither of them discusses amplification potential of BACnet.

Censys [9], Project Sonar [24], and Shodan [27] perform regular BACnet scans, finding between $5.2\,\mathrm{k}$ and $7.8\,\mathrm{k}$ less devices than our scans.

9.2 Amplification Attacks

In 2014, Rossow [25] investigated numerous UDP protocols for their susceptibility to amplification attacks. He measures amplification factors, verifies the number of available reflectors and estimates how quickly they could be harvested by a malicious actor. We add BACnet to the list of affected protocols and evaluate its potential for amplification attacks.

In 2017, Sargent *et al.* [26] discuss the amplification potential of IGMP. They find $\sim 305\,\mathrm{k}$ amplifiers with a median amplification factor of 2.4. For BACnet, we find less amplifiers but a significantly higher amplification factor.

To detect amplification attacks at the reflector network, Böttger *et al.* propose a protocol-agnostic technique based on BAF and payload entropy [4]. Krämer *et al.* present AmpPot, a honeypot designed to track amplification attacks [15].

9.3 Notification

In 2016, Li *et al.* [18] investigated the effectiveness of reporting vulnerabilities to operators. They identify $45\,770$ devices supporting at least one industrial protocol. They compare remediation rates based on communication method, verbosity, website link, translated messages. They achieve a remediation rate of about $8\,\%$. This measurable impact motivates our notification campaign. The campaign, however, was not able to achieve these high remediation rates.

10 Conclusion

We conducted multiple Internet-wide active measurements to identify $16\,485$ BACnet devices. We found that they were heavily clustered in certain ASes and prefixes. Subsequently we uncovered that $14\,\mathrm{k}$ of these devices can be

misused for amplification attacks. We evaluated the bandwidth amplification factor for a single property requested once, and a tuned payload where the same property is requested multiple times. Using this tuned payload we achieve amplification factors up to 120. We evaluated BACnet traffic in the wild and attributed the majority of it to scanning projects. Finally, we conducted a notification campaign through a CERT, observed small reductions in the number of BACnet devices, and give further advice on how to secure BACnet deployments.

Future work: We will conduct regular BACnet scans to continuously monitor the impact of our notification campaign.

Acknowledgments

The authors would like to thank the DFN-CERT for their cooperation in the vulnerability notification, Kenjiro Cho for providing the MAWI DITL data set, and Christoph Dietzel for his valuable contributions. This work has been supported by the German Federal Ministry of Education and Research, project X-CHECK, grant 16KIS0530, and project DecADe, grant 16KIS0538.

References

[1] IP2Location Geolocation DB. `https://ip2location.com`, August 2016.

[2] ASHRAE. *BACnet – A Data Communication Protocol for Building Automation and Control Systems*, 1995.

[3] ASHRAE. *BACnet – A Data Communication Protocol for Building Automation and Control Systems Addendum 135-2012aj*, 2016.

[4] Timm Böttger, Lothar Braun, Oliver Gasser, Felix von Eye, Helmut Reiser, and Georg Carle. DoS Amplification Attacks – Protocol-Agnostic Detection of Service Abuse in Amplifier Networks. In *TMA'15*.

[5] CAIDA. Routeviews Prefix to AS mapping. `www.caida.org/data/routing/routeviews-prefix2as.xml`.

[6] Nikolaos Chatzis, Georgios Smaragdakis, Jan Böttger, Thomas Krenc, and Anja Feldmann. On the Benefits of Using a Large IXP as an Internet Vantage Point. In *ACM Internet Measurement Conference*, pages 333–346. ACM, 2013.

[7] Common Vulnerabilities and Exposures. CVE-2003-0931, 11/2003.

[8] David Dittrich, Erin Kenneally, et al. The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research. *US Department of Homeland Security*, 2012.

[9] Zakir Durumeric, David Adrian, Ariana Mirian, Michael Bailey, and J. Alex Halderman. A Search Engine Backed by Internet-wide Scanning. In *SIGSAC'15*.

[10] Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. ZMap: Fast Internet-wide Scanning and Its Security Applications. In *USENIX Security'13*.

[11] Xuan Feng, Qiang Li, Haining Wang, and Limin Sun. Characterizing Industrial Control System Devices on the Internet. In *ICNP'16*.

[12] Oliver Gasser. bacnet.py: BACnet python module to parse BACnet response packets. `https://github.com/tumi8/bacnet.py`, November 2016.

[13] Oliver Gasser, Quirin Scheitle, Sebastian Gebhard, and Georg Carle. Scanning the IPv6 Internet: Towards a Comprehensive Hitlist. In *TMA'16*.

[14] IANA. IPv4 Special-Purpose Address Registry. `http://www.iana.org/assignments/iana-ipv4-special-registry/iana-ipv4-special-registry.xhtml`.

[15] Lukas Krämer, Johannes Krupp, Daisuke Makita, Tomomi Nishizoe, Takashi Koide, Katsunari Yoshioka, and Christian Rossow. AmpPot: Monitoring and Defending Against Amplification DDoS Attacks. In *RAID'15*.

[16] Brian Krebs. Hacked Cameras, DVRs Powered Today's Massive Internet Outage. `https://krebsonsecurity.com/2016/10/hacked-cameras-dvrs-powered-todays-massive-internet-outage/`, October 2016.

[17] Brian Krebs. KrebsOnSecurity Hit With Record DDoS. `http://krebsonsecurity.com/2016/09/krebsonsecurity-hit-with-record-ddos/`, September 2016.

[18] Frank Li, Zakir Durumeric, Jakub Czyz, Mohammad Karami, Michael Bailey, Damon McCoy, Stefan Savage, and Vern Paxson. You've Got Vulnerability: Exploring Effective Vulnerability Notifications. In *USENIX Security'16*.

[19] Ariana Mirian, Zane Ma, David Adrian, Matthew Tischer, Thasphon Chuenchujit, Tim Yardley, Robin Berthier, Joshua Mason, Zakir Durumeric, J Alex Halderman, et al. An Internet-Wide View of ICS Devices. In *PST'16*.

[20] H. Michael Newman. *BACnet: The Global Standard for Building Automation and Control Networks*. Momentum Press, 2013.
[21] Craig Partridge and Mark Allman. Ethical Considerations in Network Measurement Papers. *Communications of the ACM*, 2016.
[22] Ingmar Poese, Steve Uhlig, Mohamed Ali Kaafar, Benoit Donnet, and Bamba Gueye. IP Geolocation Databases: Unreliable? *ACM SIGCOMM CCR'11*.
[23] Matthew Prince. The DDoS That Almost Broke the Internet. `https://blog.cloudflare.com/the-ddos-that-almost-broke-the-internet/`, March 2013.
[24] Rapid7 Labs. Project Sonar. `https://sonar.labs.rapid7.com/`.
[25] Christian Rossow. Amplification Hell: Revisiting Network Protocols for DDoS Abuse. In *NDSS'14*.
[26] Matthew Sargent, John Kristoff, Vern Paxson, and Mark Allman. On the Potential Abuse of IGMP. *ACM SIGCOMM CCR'17*.
[27] Shodan. Map of Industrial Control Systems on the Internet. `https://icsmap.shodan.io/`.
[28] WIDE Project. MAWI Working Group Traffic Archive. `http://mawi.wide.ad.jp/mawi/` (last retrieved on 2017-09-02).

Biographies

Oliver Gasser is a scientific researcher at the Chair of Network Architectures and Services at the Technical University of Munich (TUM), Germany.

He is co-leading the Global Internet Observatory project which aims to better understand the Internet and its security by conducting Internet-wide measurements.

Oliver's research interests are empirical analysis of network security protocols such as TLS and SSH, amplification attack detection and mitigation, and more recently network scans in the IPv6 Internet.

Oliver received his M.Sc. from TUM in 2013 and is currently a PhD candidate at TUM.

Quirin Scheitle is a scientific researcher at the Chair of Network Architectures and Services at the Technical University of Munich (TUM), Germany.

He is co-leading the Global Internet Observatory project which aims to better understand the Internet and its security by conducting Internet-wide measurements.

Quirin's research interests include empirical analysis of Internet services and architectures under a security lense.

Quirin received his M.Sc. from TUM in 2012 and is currently a PhD candidate at TUM.

Benedikt Rudolph is a researcher at DE-CIX since 2016. He participates in several research projects, e.g., funded by the German Federal Ministry of Education and Research (BMBF). He actively contributes to the Internet, networking, and IXP community (e.g., RIPE, EURO-IX, DENOG).

Before joining DE-CIX he gained first practical experience as a student research assistant at Technische Universität Darmstadt, Germany, where he also received his M.Sc. in computer science with a focus on IT security.

His research interests are Internet measurements and networking technology.

Carl Denis majored in computer science with a focus on IT-security at Technical University of Munich (TUM) where he is a guest researcher.

He also pursues a doctorate at Universität der Bundeswehr and works in incident response and vulnerability handling at Siemens ProductCERT.

In his spare time he is concerned with secure and automated infrastructures.

Nadja Schricker is currently studying Computer Science at the Technical University of Munich.

She recently finished her Bachelor's Thesis on the topic "Active Security Evaluation with Network Scans".

Georg Carle is professor at the Department of Informatics of the Technical University of Munich, holding the Chair of Network Architectures and Services.

He studied at University of Stuttgart, Brunel University, London, and Ecole Nationale Superieure des Telecommunications, Paris.

He did his PhD in Computer Science at University of Karlsruhe, and worked as postdoctoral scientist at Institut Eurecom, Sophia Antipolis, France, at the Fraunhofer Institute for Open Communication Systems, Berlin, and as professor at University of Tübingen.