THE UNIVERSITY OF UTAH

An In-Depth Analysis on Adoption of Attack Mitigations in Embedded Devices

Ruotong Yu[†,★], Francesca Del Nin[¶], Yuchen Zhang[★], Shan Huang[★], Pallavi Kaliyar[μ], Sarah Zakto[‡], Mauro Conti [¶,£], Georgios Portokalidis [★], Jun Xu[†,★]

† THE UNIVERSITY OF UTAH

★ STEVENS INSTITUTE of TECHNOLOGY THE INNOVATION UNIVERSITY

‡ CYBER -ITL-

¶ UNIVERSITÀ DEGLI STUDI DI PADOVA

μ NTNU Norwegian University of Science and Technology

£ TU Delft Delft University of Technology

LASER' 2022

THE UNIVERSITY OF UTAH

Background

Embedded Systems Are Everywhere

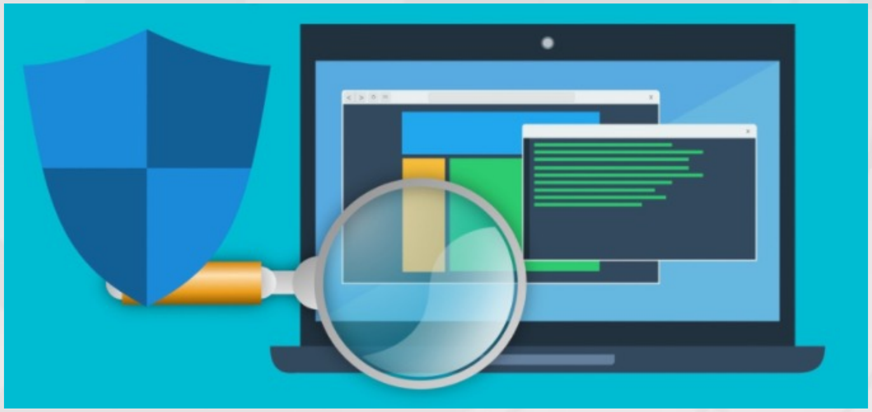
THE UNIVERSITY OF UTAH

Background

User-space Mitigations

- *Stack Canary*
- *Non-executable Stack (NX)*
- *Address Space Layout Randomization (ASLR)*

....



Kernel-Level Mitigations

- *Stack Protector*
- *Kernel-level Address Space Layout Randomization (KASLR)*

....

Mitigations are used to Protect Desktop Systems

THE UNIVERSITY OF UTAH

Motivations

Mitigations Are Missing

Brand	Device	Count	ASLR %	NX %	RELRO %	Canary %	CPU
Ubuntu Desktop	16.04	5379	23.12	99	100	79.43	x86
Asus	rt-ac55u	334	0	0	1.8	0	MIPS
D-LINK	dir-850l	118	0	0	3.39	0	MIPS
Linksys	e2500	201	8.79	0	3.48	0	ARM

Table. Adoption rates of user-space mitigations from popular home routers (https://cyber-tl.org/assets/papers/2018/build_safety_of_software_in_28_popular_home_rouers.pdf)

THE UNIVERSITY OF UTAH

Research Question

Q1: With all the needed support available, do embedded devices adopt attack mitigations?

Q2: Is the adoption of the attack mitigations improving over time?

Q3: If the attack mitigations are missing? What are the possible reasons?

Perform a large-scale study on evaluating the mitigation adoption on embedded devices

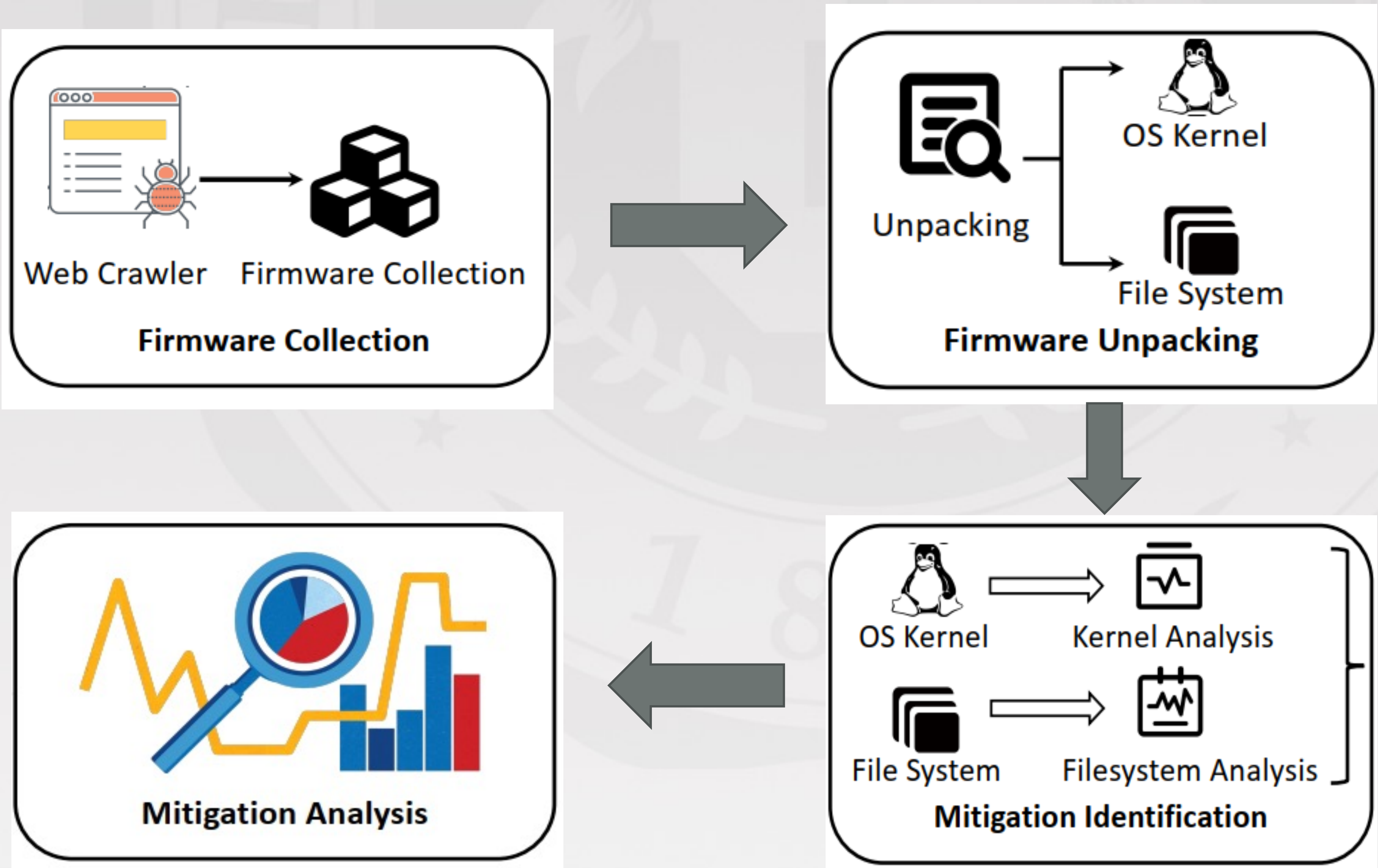
THE UNIVERSITY OF UTAH

Challenges with Large Scale Analysis

- Building High-quality Dataset
  - Previous datasets are outdated and even invalid today
- Unpacking Firmware Images
  - Firmware images are organized in diverse formats
  - Raw data format kernel cannot be fully recovered
- Identifying Attack Mitigations
  - Existing tools like Hardening-Check. Checksec have design limitations
  - Kernel mitigations are rarely considered by the tools

THE UNIVERSITY OF UTAH

Approach to Large-Scale Analysis



THE UNIVERSITY OF UTAH

Data Collection

- Web Crawler

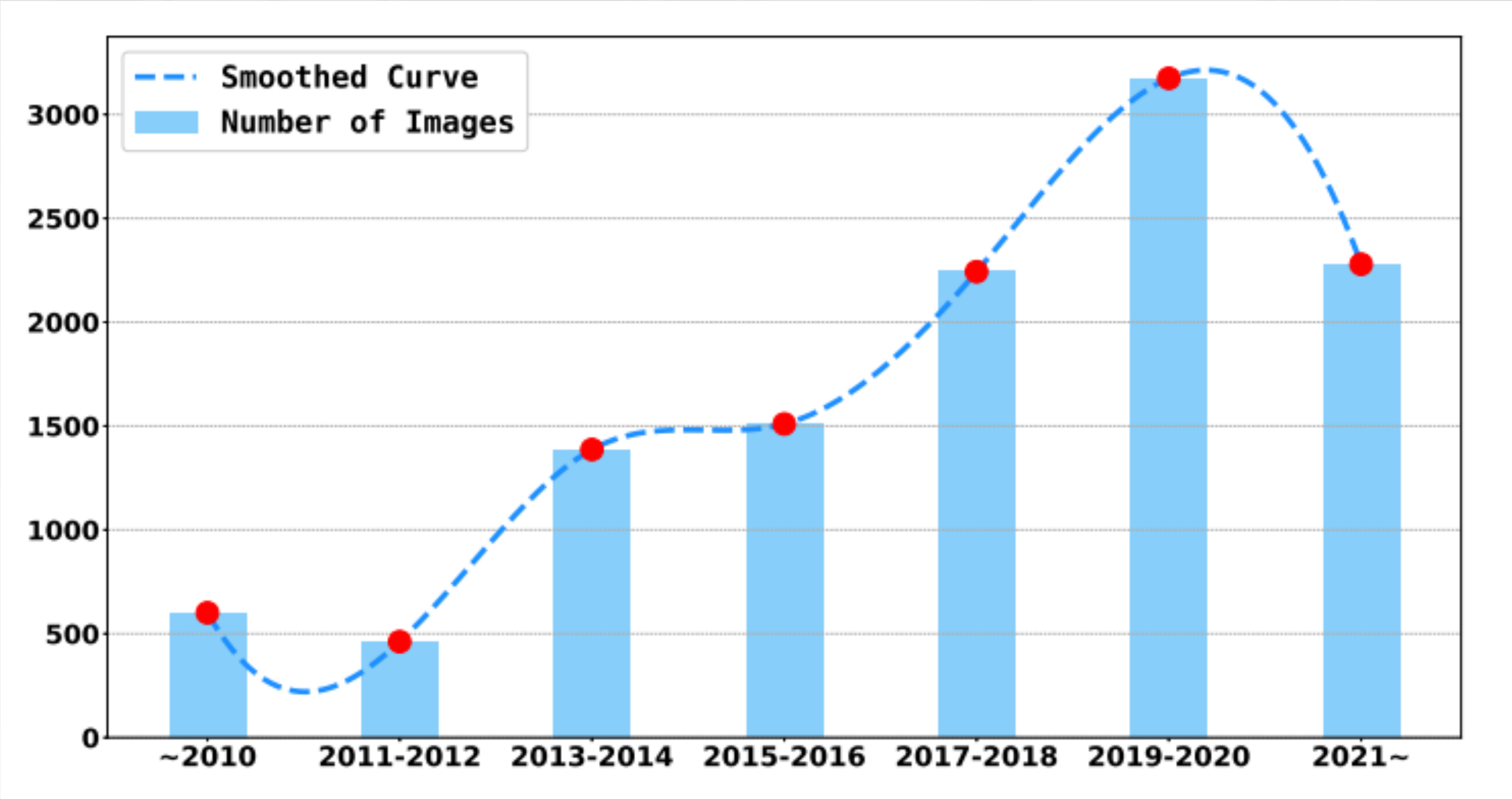
  - Previous dataset are invalid or outdated today
    1. "A large-scale analysis of the security of embedded firmware" USENIX Security 2014 (Only 5% URLs are valid)

  - Previous work has designed web crawler for the same purpose, but need to be updated
    1. "Towards automated dynamic analysis for Linux-based embedded firmware" NDSS 2016 (Only few crawler working properly)

Update the web crawler from FIRMADYNE

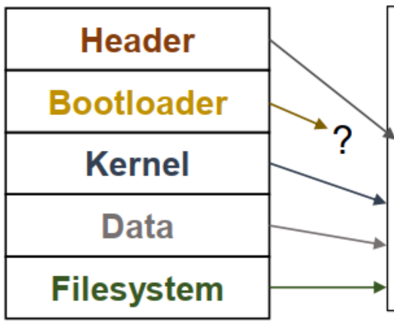
THE UNIVERSITY OF UTAH

Data Collection Result



- ***In total, we collected over 18,000 firmware images from 38 vendors. The firmware range from 1998 to 2021 and include most common types of devices.***

Firmware Unpacking




```
DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             uImage header, header size: 64 bytes,
                              header CRC:0xC932233,  image size: 2692516 ...
64            0x40            Linux kernel ARM boot executable zImage ...
16636         0x40FC          gzip compressed data, maximum compression ...
2752512       0x2A0000        JFFS2 filesystem, little endian
```

Structure of firmware images

BINWALK output for `linksys-EA4500-2.1.42.183584_prod.img`

Extract Filesystem

- Search for standard directories like bin, sbin, lib and etc
- The directories will then recursively traversed to identify binaries

THE UNIVERSITY OF UTAH

Firmware Unpacking

Extract Linux Kernel

- Improve signatures inherited from Binwalk to extract kernel
- Use vmlinux-to-elf[1] tool to recover the Linux kernel into ELF format

Linux version 2.6.28 (arica@localhost.localdomain) (gcc version 4.4.0 (Faraday C/C++ Compiler Release 20100325)) #72 PREEMPT Wed Apr 29 18:49:51 CST 2015

A. String recognized

Linux version 4.4.35_hi3796mv200 (xushaohui@raysharp-PowerEdge-R720) Linux version 4.14.221 (builder@buildhost)

B. String not recognized

[1] marin-m. vmlinux-to-elf. https://github.com/marin-m/vmlinux-to-elf, 2021

THE UNIVERSITY OF UTAH

Firmware Unpacking Result

Vendor	# of Images	Filesystems		Linux Kernels		
		Total	ELF (k)	Total	.config	converted
Cerowrt	2	2	0.4	0	0	0
Haxorware	2	1	0.2	0	0	0
AT&T	4	4	0.6	0	0	0
360	5	4	0.5	4	0	2
Actiontec	6	5	0.4	0	0	0
Buffalo	6	4	0.5	4	0	2
Camius	6	6	0.5	6	0	6
GOCloud	8	7	0.9	0	0	0
Phicomm	13	8	1.9	3	1	3
ZyXEL	15	7	0.8	7	0	3
CenturyLink	18	7	0.8	2	0	1
Polycom	21	0	0	16	16	0
u-blox	31	0	0	0	0	0
TENVIS	41	25	0.9	31	0	0
MikroTik	49	32	4.3	0	0	0
Foscam	83	0	0	10	0	0
AVM	107	22	5.0	0	0	0
RouterTech	144	143	25.8	142	0	0
Belkin	165	60	7.9	60	0	33
Linksys	166	74	17.1	101	24	75
Mercury	169	27	1.5	27	0	27
Supermicro	187	5	1.3	187	7	9
Digi	214	3	1.5	5	1	2
NETCore	255	152	10.2	138	1	85
Moxa	400	107	32.0	0	0	0
TRENDnet	409	142	15.3	158	3	70
Tenda	467	252	33.6	142	0	118
Ubiquiti	512	479	204.7	449	59	436
QNAP	576	297	296	0	0	0
Hikvision	607	0	0	190	41	186
Synology	672	671	1375.4	0	0	0
TomatoShibby	692	692	127.8	314	0	23
Tp-Link-zh	992	464	65.7	385	53	325
ASUS	1,099	1,069	273.2	438	54	288
D-Link	1,172	86	15.9	116	11	92
Tp-Link-en	1,186	654	76.3	565	43	544
NETGEAR	3,682	980	173.9	1,293	269	957
OpenWrt	3,837	2,546	191.2	3,184	0	0
Total	**18,020**	**9,037**	**2,964**	**7,977**	**581**	**3,287**

- ***We unpacked 10,685 out of 18,020 firmware images with success rate 59.3%.***
- ***In summary, we collected 9,037 filesystems with over 3,000k ELF binaries and 7,977 Linux kernels***
- ***In among of 7,997 Linux kernel, we found 581 of them containing .config file and 3,287 converted to ELF format***

THE UNIVERSITY OF UTAH

User-space Mitigation

Stack Canary
- Search __stack_chk_fail in symbols

Fortify Source
- Search indicator functions (e.g strcpy_chk) in symbols

Position-Independent Code (PIC/PIE)
- Check type in program header (ET_DYN)

Relocation Read-Only (RELRO)
- Check permission flag of .got and .got.plt section

Non-executable Stack (NX Stack)
- Check presence of PT_GNU_STACK in program header

THE UNIVERSITY OF UTAH

Improvement of Traditional Approach

Stack Canary

- For dynamically linked binary, search __stack_chk_fail in symbols
- For statically linked binary, search error message "***stack smashing detected***"

Fortify Source

- For dynamically linked binary, search indicator functions (e.g strcpy_chk) in symbols
- For statically linked binary, search error message "***buffer overflow detected***"

Relocation Read-Only (RELRO)

- Check permission flag of .got and .got.plt section
- Flags (**BIND_NOW, DT_BIND_NOW** and etc) are used to determine full RELRO

THE UNIVERSITY OF UTAH

Kernel-level Mitigation

Attack Vector	Mitigation	Building Configuration	Release Version	First Release
Stack Overflow	Stack Protector	CONFIG_HAVE_CC_STACKPROTECTOR	ARM:v2.6 MIPS:v3.11 PowerPC:4.20	2009
Privilege Escalation	PXN[2]	–[1]	ARM:v3.19 AArch64:v3.7	2012
Control Flow Hijacking	KASLR	CONFIG_RANDOMIZE_BASE	ARM:v4.6 MIPS:v4.7 PowerPC:v5.2	2014
Heap Corruption	Freelist Randomization	CONFIG_SLAB_FREELIST_RANDOM	v4.7	2016
Information Leakage	USERCOPY	CONFIG_HARDENED_USERCOPY	v4.8	2016
Buffer Overflow	Fortify Source	CONFIG_Fortify_Source	AArch64&PowerPC:v4.13, ARM-32:v4.17, MIPS:v5.5	2017
Code Injection	Non-executable Memory	CONFIG_STRICT_KERNEL_RWX	ARM:v4.11 PowerPc:v4.13 (MIPS does not support)	2017

[1] "–" indicates the mitigation is not affected by the building configuration.
[2] x86/x64 have similar mitigations called SMEP and SMAP. They are not considered because no x64/x86 kernels are identified in our dataset.

Table. Memory Related Attack mitigations in Linux Kernel

Rules:

1. ***Active in Linux distributions***
2. ***Released over three years***
3. ***Applicable to deployed systems***

Kernel-level Mitigation Identification

Kernel Version

- Kernel version is available in both .config file and string constant

Build Configuration

- Only when the option is present and its value is “=y”, it’s enabled

ELF Format Kernel

- Use indicator functions from recovered ELF kernel (__stack_chk_fail for Stack Protector and etc)

THE UNIVERSITY OF UTAH

User-space Evaluation Approach

Experiment to Answer Q1

- Measure the mitigation adoption rate for all the embedded binaries
- Breakdown the binaries by types

Experiment to Answer Q2

- Keep track of mitigation change over time
- Evolution of individual firmware

Experiment to Answer Q3

- Understand the limitation of building tool
- Evaluate reused binaries
- Measure mitigation overhead
- Case study on embedded vulnerabilities

THE UNIVERSITY OF UTAH

Kernel-Level Evaluation Approach

Experiment to Answer Q1

- Measure the mitigation adoption rate for all the Linux kernel
- No further analysis as kernels are barely protected

Experiment to Answer Q2

- Keep track of mitigation change over time on Stack Protector
- Measure the gap between the release time and building time of kernels

THE UNIVERSITY OF UTAH

User-space Findings to Answer Q1

Q1: Do embedded devices adopt attack mitigations?

- ***The adoption rates of mitigations by embedded binaries are surprisingly low***

  85.3% binaries protected with Stack Canary on desktop but the number drop to 29.7% on embedded system

- ***The adoption rates of mitigation dramatically vary across vendors***

  Best performance vendor achieve 81.5% Stack Canary but worst performance vendors completely ignore it

Vendors	ELF (k)	Canary	RELRO	NX	Fortify	PIE
Haxorware	0.2	0	0	0	0	14.9
Actiontec	0.4	0.5	0	47.2	0.5	13.4
Cerowrt	0.4	0	0	0	0	9.8
360	0.5	60.0	0	0	0	8.9
Buffalo	0.5	0	0	45.8	0	6.0
Camius	0.5	11.9	0	92.1	1.3	11.9
AT&T	0.6	0	0	0	0	6.3
CenturyLink	0.8	0	0	0	0	0.6
Zyxel	0.8	1.0	0	97.3	0.9	11.6
GOCloud	0.9	0	0	98.2	0	14.9
TENVIS	0.9	0	0	0	0	34
Supermirco	1.3	19.4	3.2	97.8	16.1	18.5
Digi	1.5	0	0	3.5	0	18.5
Mercury	1.5	0	0	0	0	31.5
Phicomm	1.9	0.1	0.8	21.2	0	47.2
MikroTik	4.3	0.2	7.9	81.0	0.07	5.8
AVM	5.0	81.5	89.4	95.6	0.04	90.8
Belkin	7.9	0.2	3.8	7.4	1.6	11.0
NETCore	10.2	11.3	0.02	0.06	0.2	16.4
TRENDnet	15.3	0.4	0.3	10.1	0.05	13.6
Dlink	15.9	0.4	0.4	30.4	0.04	9.1
Linksys	17.1	0.5	3	60.4	0.8	9.0
RouterTech	25.8	0	0	0	0	15.0
Moxa	32.0	39.3	15.0	75.7	35.5	31.8
Tenda	33.6	0.6	2.3	30.5	0.01	11.7
Tp-Link-zh	65.7	2.9	0.4	38.7	0.1	18.3
Tp-Link-en	76.3	0.5	0.9	36.6	0.6	21.5
TomatoShibby	127.8	0.1	1.0	23.2	0	8.4
NETGEAR	173.9	2.2	4.4	55.9	0.5	11.4
OpenWrt	191.2	0	0	99.9	0	0
Ubiquiti	204.7	6.7	1.0	15.6	25.0	9.5
ASUS	273.2	1.3	1.4	46.8	0.05	8.3
QNAP	296.0	80.1	3.1	99.2	1.4	7.7
Synology	1375.4	43.6	36.7	99.5	43.5	13.5
Ave (Vendor)	87.2	10.7	5.2	41.5	3.5	16.5
Ave (Binary)	-	29.7	18.3	76.2	22.5	11.6
Debian	34.0	85.3	98.1	99.7	55.6	94.0

THE UNIVERSITY OF UTAH

More Findings by Breakdown Binaries

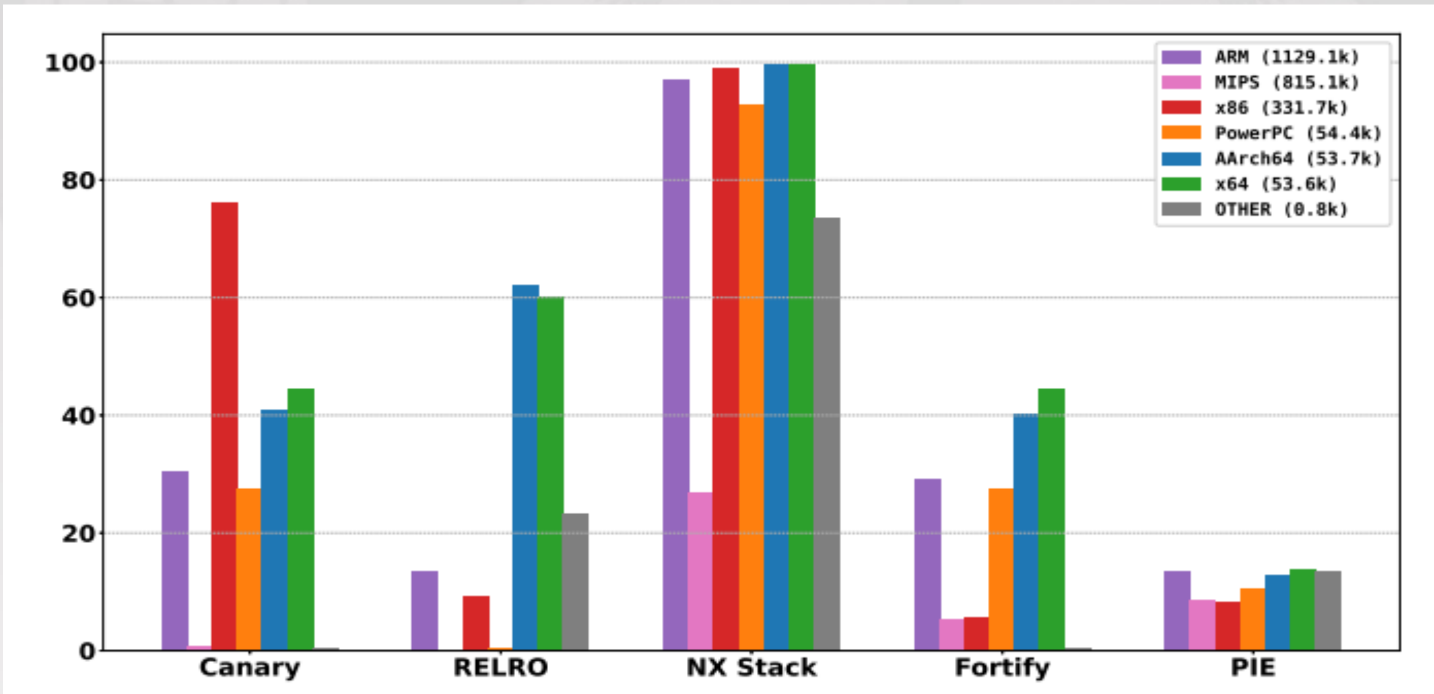


Fig. Adoption rates of user-space mitigations by binaries running on different architectures.

MIPS as the second largest group has the lowest adoption rates in nearly every mitigation

In comparison, x86/AArch64 binaries have relatively higher adoption of mitigations.

THE UNIVERSITY OF UTAH

Kernel-level Findings to Answer Q1

Category	Total	Stack Protector	PXN	KASLR	FreeList	Usercopy	Fortify	Kernel RWX
Analyzed	3,347	2,831	839	2,062	2,063	1,980	525	564
Unsupported	-	2,078	798	2,048	2,049	1,968	521	555
Protected	-	159	41	0	0	3	4	9

Table. Adoption result of kernel-level mitigations

Kernel-level mitigations are rarely adopted in embedded devices

THE UNIVERSITY OF UTAH

User-space Findings to Answer Q2

Q2: Is the adoption of the attack mitigations improving over time?

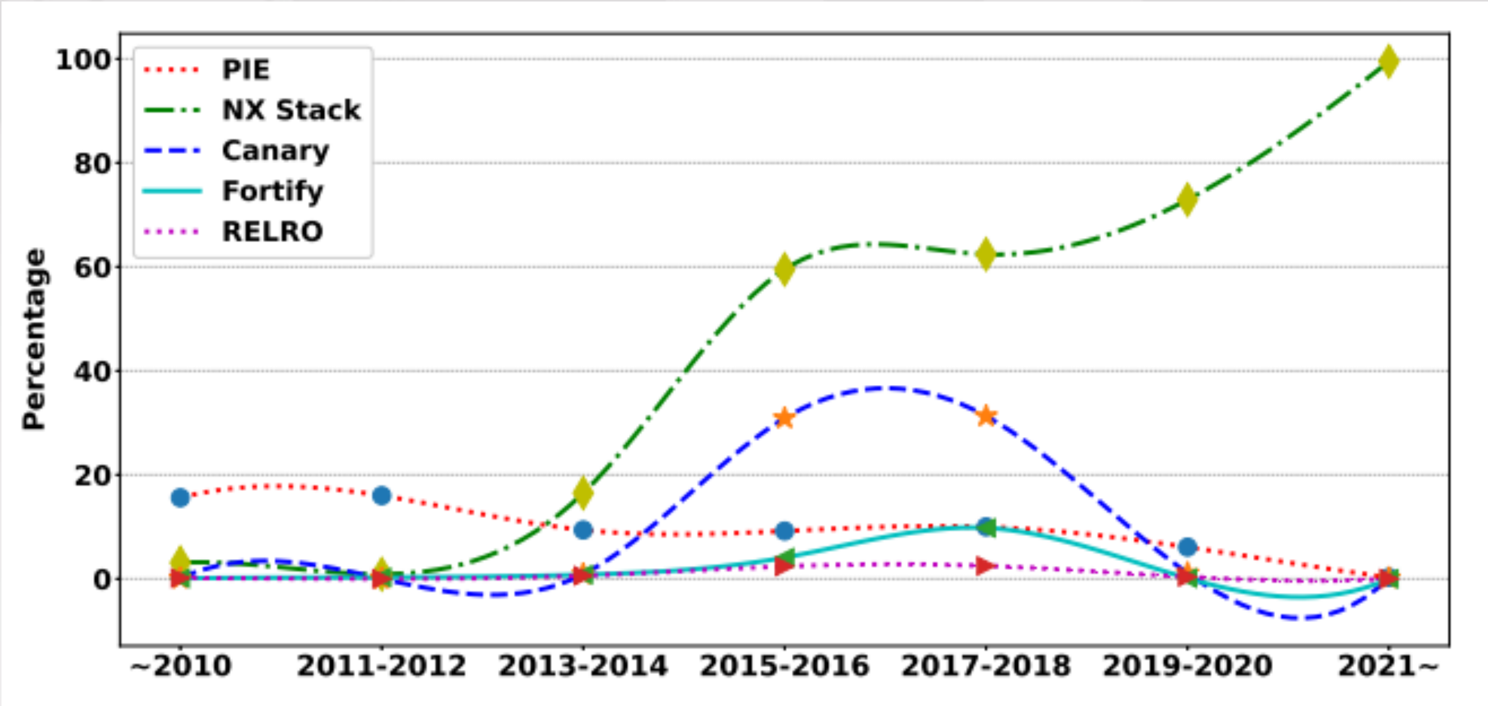


Fig. Adoption rates of user-space mitigations across time.

Only adoption of NX Stack presents a positive trend

THE UNIVERSITY OF UTAH

User-space Findings to Answer Q2

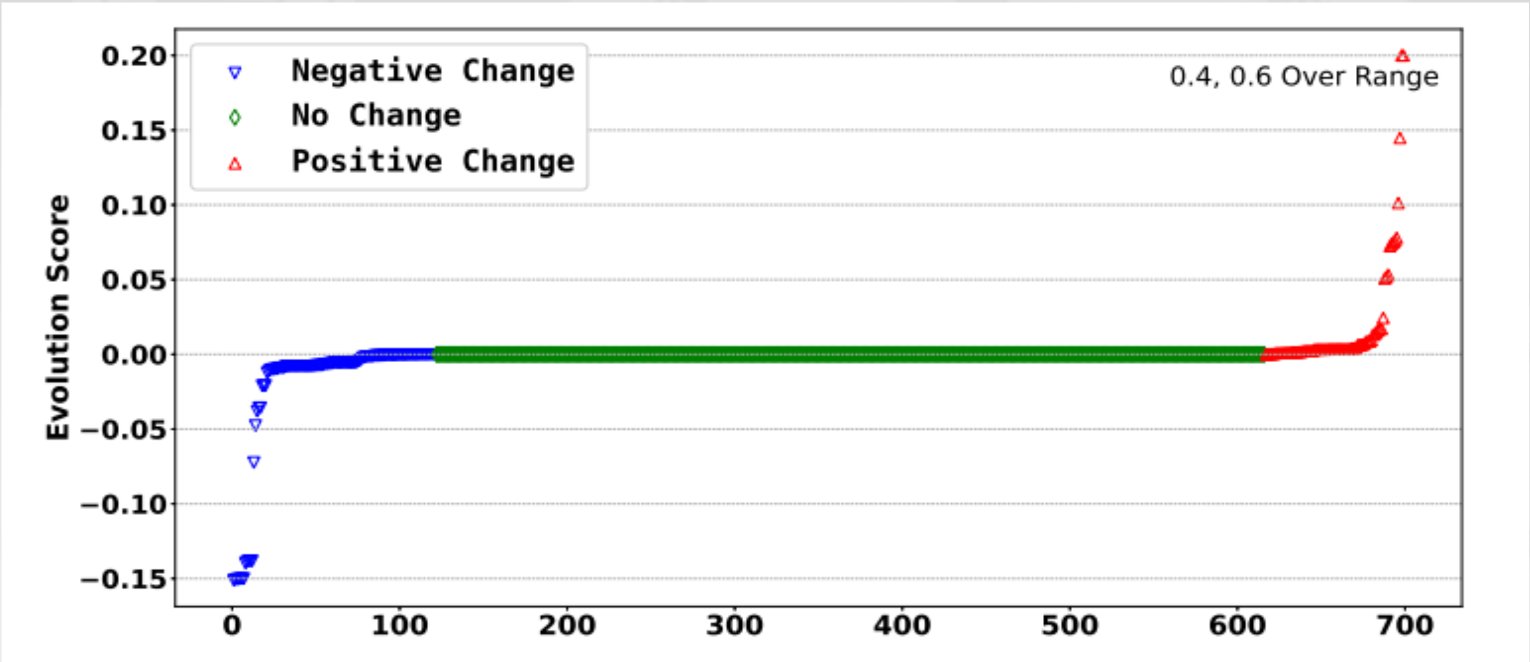


Fig. Evolution score of individual firmware in the adoption of Stack Canaries. Each point represents a firmware with multiple version.

Most of the firmware present no change

The evidence shows that the adoption of user-space mitigations is not improving

Kernel-level Findings to Answer Q2

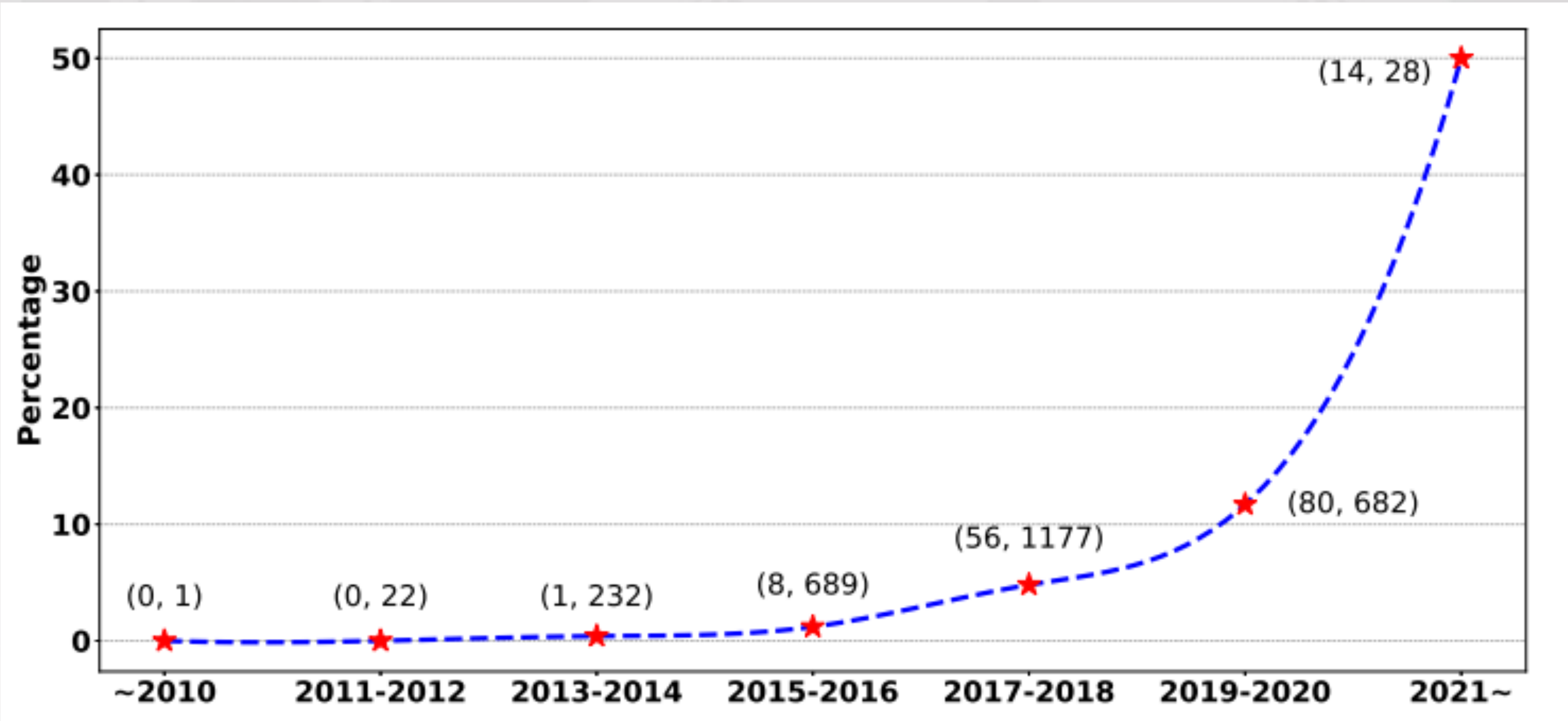


Fig. Evolution of Stack Protector across time

The adoption rate of Stack Protector consistently increase over the past decade

THE UNIVERSITY OF UTAH

Findings to Answer Q3

Q3: What are the possible reasons of missing mitigation?

Version	Default Kernel	Canary	SC[1] Dependency	RELRO	Fortify	PIE
2021-02	v5.10	✓	✓	✓	✓	✓
2020-11	v5.4	✓	✓	✓	✓	✓
2019-11	v4.19	✓	✓	✓	✓	✓
2018-11	v4.16	✓	✓	✓	✓	☒
2017-11	v4.13	✓	✓	☒	☒	☒
2016-11	v4.8	✓	✓	☒	☒	☒
2015-11	v4.3	✓	✓	☒	☒	☒
2014-11	v3.17	✓	✓	☒	☒	☒
2013-11	v3.11	✓	✓	☒	☒	☒
2012-11	v3.6	✓	☒	☒	☒	☒
2011-11	v3.1	✓	☒	☒	☒	☒
2010-11	v2.6	✓	☒	☒	☒	☒
2009-11	v2.6	✓	☒	☒	☒	☒

[1] "SC" is short for Stack Canaries.

Table. Availability of attack mitigations in different versions of Buildroot

Restrictions of Building Tools

THE UNIVERSITY OF UTAH

Findings to Answer Q3

Ratio (%)	Unique	Total	Vendor
11.1	0.1	0.9	TENVIS
100	0.2	0.2	Haxorware
100	0.2	0.2	Cerowrt
50.0	0.2	0.4	AT&T
33.3	0.2	0.6	Camius
60.0	0.3	0.5	GOCloud
33.3	0.3	0.9	Actiontec
75.0	0.3	0.4	Buffalo
80.0	0.4	0.5	360
100	0.5	0.5	Phicomm
26.3	0.5	1.9	Mercury
33.3	0.5	1.5	Zyxel
87.5	0.7	0.8	CenturyLink
60.0	0.9	1.5	Digi
60.0	0.9	1.5	Supermirco
36.0	1.8	5.0	AVM
48.8	2.1	4.3	MikroTik
1.5	2.9	191.2	OpenWrt
43.1	3.4	7.9	Belkin
36.3	3.7	10.2	NETCore
15.5	4.0	25.8	RouterTech
29.6	4.7	15.9	Dlink
17.3	5.8	33.6	Tenda
4.5	5.8	127.8	TomatoShibby
40.9	7.0	17.1	Linksys
50.9	7.8	15.3	TRENDnet
40.0	12.8	32.0	Moxa
4.3	12.8	296.0	QNAP
24.1	15.8	65.7	Tp-Link-zh
10.1	20.5	204.7	Ubiquiti
30.9	23.6	76.3	Tp-Link-en
16.8	29.2	173.9	NETGEAR
10.9	29.7	273.2	ASUS
4.7	64.9	1375.4	Synology
8.9	7.8	87.2	**Average**

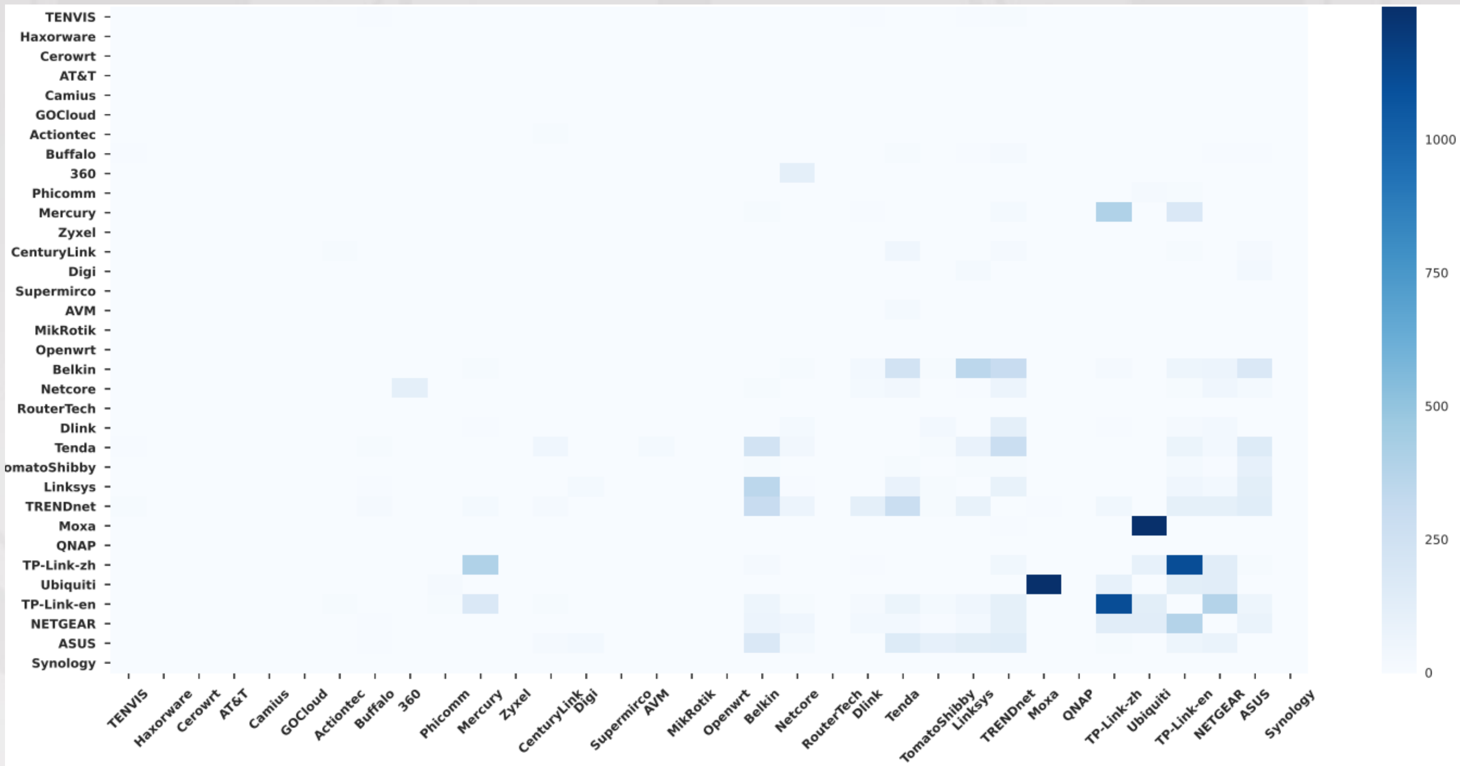


Fig. Heatmap showing the binaries vendors borrow from each other

Massive Reuse of Binaries

Potential Reasons for Q3

Overhead	NX	Canary	PIE	RELRO	Fortify
Storage	0	6.7%	11.5%	17.3%	17.3%
Memory	0	0	0	0	0
Runtime	0	6.6%	8.45%	10.7%	10.9%

Table. Cost of attack mitigations on SPEC CPU2006.

Mitigations like Stack Canary, PIE and RELRO have observable overhead.

THE UNIVERSITY OF UTAH

Potential Reasons for Q3

CVE-2021-35392 ('WiFi Simple Config' stack buffer overflow via UPnP)
CVE-2021-35393 ('WiFi Simple Config' heap buffer overflow via SSDP)

CVEs affected Realtek SDK. Reported on 2021

- ***memory corruption vulnerabilities are common on embedded devices***

Question: *Are the adoption rates higher on devices containing more vulnerabilities?*

- ***Vulnerable binaries present no broader adoption of the attack mitigations***

THE UNIVERSITY OF UTAH

How previous work motivated us?

Data Collection

- Extend web crawler based on previous research
- Reuse state-of-art firmware unpacking tools

Mitigation Identification

- Improve the user-space mitigation identification approach
- Added kernel level mitigation approach

THE UNIVERSITY OF UTAH

Any intermediate result?

Raw Data

- We keep all the firmware images, filesystems, Linux Kernels

Statical Result

- We save the mitigation adoption information for each binary as running mitigation identification for millions of binary is time consuming

THE UNIVERSITY OF UTAH

Do we share the data?

Yes, we share all the dataset we collected

- We share the download links for the firmware images and the metadata

We did not report any of our findings to the vendor

- We did not directly contact the vendors or use any private data for our evaluation

Limitations

- Imbalance of Dataset
  - Not every vendor has the same amount of data involved
  - The data samples are not evenly distributed over time
- Reliability of Mitigation Identification
  - Obfuscation will affect our identification of attack
  - Encoding strings or destroying symbols may influence our result
  - Static approaches itself have limitations

THE UNIVERSITY OF UTAH

Thank You for Listening!