Information Security Research and Education

N. Asokan
Twitter: @nasokan, WWW: https://asokan.org/asokan
About me

Professor, Aalto University, from Aug 2013
Professor, University of Helsinki, 2012-2017

Associate Editor-in-Chief, IEEE Security & Privacy

Previously
  Nokia (14 y; built up Nokia security research team)
  IBM Research (3 y)

More information on the web (https://asokan.org/asokan) or Twitter (@nasokan)
Secure Systems Group

Prof N. Asokan
Professor, Department of Computer Science
Director: Helsinki-Aalto Center for Information Security
https://asokan.org/asokan/

Prof Tuomas Aura
Professor, Department of Computer Science
https://people.aalto.fi/tuomas_aura

Dr Andrew Paverd
Research Fellow, Department of Computer Science
Deputy Director: Helsinki-Aalto Center for Information Security
https://ajpaverd.org
Secure Systems Group: Mission

How to make it possible to build systems that are simultaneously easy-to-use and inexpensive to deploy while still guaranteeing sufficient protection?
Secure Systems Group

In Asokan’s projects:
  • 3 postdocs
  • 5 full-time + 3 part-time PhD students

Several MSc students
  • Best InfoSec thesis in Finland 2017, 2016 & 2014, Tietoturva ry
  • Runner-up for Best CS thesis in Finland 2014, TKTS ry

Projects funded by
  • Academy of Finland, Tekes
  • Direct industry support: E.g., Intel http://www.icri-sc.org, [NEC Labs, Huawei]

http://cs.aalto.fi/secure_systems/
Aalto University

Established in 2010, named in honour of Alvar Aalto, the famous Finnish architect.

Science and art meet technology and business.
Promoting entrepreneurship

70 to 100 Companies are founded every year in our ecosystem

Entrepreneurship is a more popular career option than ever – in the last four years, over 2 000 students have studied through the Aalto Ventures Program

MIT Skolltech initiative rated Aalto’s innovation ecosystem among the top-5 rising stars in the world

50% of Finnish startups that originate from universities come from the Aalto community
NOBODY IN THEIR RIGHT MIND WOULD COME TO HELSINKI IN NOVEMBER.
Research

Building systems that are secure, usable, and deployable
Current themes: Platform Security

How can we design/use pervasive hardware and OS security mechanisms to secure applications and services?

HardScope: Thwarting DOP with Hardware-assisted Run-time Scope Enforcement

Thomas Nyman, Ghada Dessouky, Shaza Zeitouni, Aaro Lehikoinen, Andrew Paverd, N. Asokan, Ahmad-Reza Sadeghi

(Submitted on 29 May 2017)

The widespread use of memory unsafe programming languages (e.g., C and C++), especially in embedded systems and the Internet of Things (IoT), leaves many systems vulnerable to memory corruption attacks. A variety of defenses have been proposed to mitigate attacks that exploit memory errors to hijack the control flow of the code at run-time, e.g., (fine-grained) ASLR or Control Flow Integrity (CFI). However, recent work on data-oriented programming (DOP) demonstrated the possibility to construct highly-expressive (Turing-complete) attacks, even in the presence of these state-of-the-art defenses. Although multiple real-world DOP attacks have been demonstrated, no suitable defenses are yet available.

We present run-time scope enforcement (RSE), a novel approach designed to mitigate all currently known DOP attacks by enforcing compile-time memory safety constraints (e.g., variable visibility rules) at run-time. We also present HardScope, a proof-of-concept implementation of hardware-assisted RSE for the new RISC-V open instruction set architecture. We demonstrate that HardScope mitigates all currently known DOP attacks at multiple points in each attack. We have implemented HardScope in hardware on the open-source RISC-V Pulino microcontroller. Our cycle-accurate simulation shows a real-world performance overhead of 7.1% when providing complete mediation of all memory accesses.

https://arxiv.org/abs/1705.10295
Current themes: Platform Security

Enabling developers to secure apps/services using h/w and OS security
Example: SafeKeeper – using Intel SGX on server-side to protect passwords

Over 560 Million Passwords Discovered in Anonymous Online Database

https://ssg.aalto.fi/projects/passwords/
Can we guarantee performance of machine-learning based systems even in the presence of adversaries?

[Image of an article from IEEE Transactions on Computers, Vol. 66, No. 10, October 2017]

Off-the-Hook: An Efficient and Usable Client-Side Phishing Prevention Application

Samuel Marchal, Member, IEEE, Giovanni Armano, Tommi Gröndahl, Kalle Saari, Nidhi Singh, and N. Asokan, Fellow, IEEE

Abstract—Phishing is a major problem on the Web. Despite the significant attention it has received over the years, there has been no definitive solution. While state-of-the-art solutions have reasonably good performance, they suffer from several drawbacks including potential to compromise user privacy, difficulty of detecting phishing websites whose content change dynamically, and reliance on features that are too dependent on the training data. To address these limitations we present a new approach for detecting phishing websites in real-time as they are visited by a browser. It relies on modeling inherent phishing limitations stemming from the constraints they face while building a webpage. Consequently, the implementation of our approach, Off-the-Hook, exhibits several notable properties including high accuracy, brand-independence and good language-independence, speed of decision, resilience to dynamic phishing and resilience to evolution in phishing techniques. Off-the-Hook is implemented as a fully-client-side browser add-on, which preserves user privacy. In addition, Off-the-Hook identifies the target website that a phishing webpage is attempting to mimic and includes this target in its warning.

We evaluated Off-the-Hook in two different user studies. Our results show that users prefer Off-the-Hook warnings to Firefox warnings.

https://ssg.aalto.fi/projects/phishing/
Current themes: Machine Learning & Security

Applying ML for Security & Privacy problems; Security & Privacy concerns in ML
Example: MiniONN – privacy-preserving neural network predictions


https://eprint.iacr.org/2017/452

MiniONN (ACM CCS 2017)
Current themes: Emerging topics

Distributed consensus and blockchains (theory, applications) [AoF project BCon, ICRI-SC]
- Can hardware security mechanisms help design scalable consensus schemes?

Securing IoT (scalability, usability) [AoF project SELIoT]
- How do we secure IoT devices from birth to death?

Security and privacy of vehicle-to-X (V2X) communication [ICRI-SC]
- How to reconcile privacy and lawful interception?

Stylometry and security [HICT scholarship]
- Can text analysis help detect deception?
ICRI-SC

Intel Collaborative Research Institute for Secure Computing
• Only Intel Institute for security outside the US

ICRI-SC for mobile and embedded systems security
• 2012-2017 (Aalto, TU Darmstadt, UH; Aalto joined in 2014)
• Nearly 1 M€ invested in Aalto and UH

ICRI-CARS for autonomous systems security
• 2017-2020 (Aalto, TU Darmstadt, RU Bochum, U Luxembourg, TU Wien)
Media coverage of our research
Education

Training the next generation of information security researchers and professionals
Master's Programme in Computer, Communication and Information Sciences - Security and Cloud Computing

Acquire a world-class education in information security at Aalto University!

Studies in Security and Cloud Computing give students a broad understanding of the latest and future technologies for secure mobile and cloud computing systems. Students will gain both practical engineering knowledge and theoretical insights into

- secure systems engineering,
- distributed application development.

Degree:
Master of Science (Technology)
More information.

ECTS:
120 ECTS

Field of Study:
Technology and Engineering

Duration:
2 years, full-time

Eligibility:
An appropriate Bachelor's degree or an equivalent qualification.

Tuition fees & scholarships:
Yes, for non-EU citizens.
More information

Language of Instruction:
English
More information.

Organising schools:
School of Science

Application period:
2017-12-15 - 2018-03-24

SECCLO
Master’s Programme in Security and Cloud Computing
(Erasmus Mundus)

Applications: 4.12.2017 – 17.01.2018

~20 scholarships

secclo.aalto.fi  secclo@aalto.fi  facebook.com/secclo
Helsinki-Aalto Center for Information Security (HAIC)

Joint initiative: Aalto University and University of Helsinki

Mission: attract/train top students in information security
  • Offers financial aid to top students in both CCIS Security and Cloud Computing & SECCLO
  • Three HAIC scholars in 2017; Five (expected) in 2018

Supported by industry donations
  • F-Secure, Intel, Nixu (2017)
  • F-Secure, Huawei (2018)

Targeted donations possible

https://haic.aalto.fi/
InfoSec Research and Education @ Aalto

2014
- ACM CCS (1)
- Proc. IEEE (1)
- WWW (1)
- PerCom (1)
- ASIACCS (1)
- Black Hat USA (1)

Runner-up: Best CS MSc Thesis in Finland
20+ MSc and BSc theses yearly

2015
- ACM CCS (2)
- ACM WiSec (1)
- PerCom (1)
- Black Hat Europe (1)
- ASIACCS (1)
- UbiComp (1)

2016
- ACM CCS (1)
- NDSS (2)
- ICDCS (1)
- CeBIT (1)
- Black Hat Europe (1)

Best InfoSec MSc thesis in Finland

2017
- ACM CCS (1)
- DAC (1)
- ICDCS (2)
- RAID (1)
- IEEE IC (1)
- IEEE TC (1)

Best InfoSec MSc thesis in Finland

2018
- CT-RSA (1)
- Euro S&P (1)
Information Security Research and Education

N. Asokan
Twitter: @nasokan, WWW: https://asokan.org/asokan
Machine Learning
in the presence of adversaries

(joint work with Mika Juuti, Jian Liu, Andrew Paverd and Samuel Marchal)
Machine Learning is ubiquitous

The ML market size is expected to grow by **44% annually** over next five years.
In 2016, companies invested up to **$9 Billion** in AI-based startups.

Machine Learning for security/privacy

Access Control

https://ssg.aalto.fi/projects/phishing/

Deception Detection
Security & privacy of machine learning
Which class is this?  
**School bus**

Which class is this?  
**Ostrich**


Skip to robust adversarial examples
Which class is this? Building

Which class is this? Ostrich

Which class is this?  
**Panda**

Which class is this?  
**Gibbon**

Which class is this? Cat

Which class is this? Desktop computer

DolphinAttack: Inaudible Voice command

Guoming Zhang  Chen Yan  Xiaoyu Ji
Tianchen Zhang  Taimin Zhang  Wenyuan Xu

Zhejiang University

ACM CCS 2017
Fredrikson et al. “Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures”, ACM CCS ’15. 
A more realistic Machine Learning pipeline

Where is the adversary?
Malicious data owner

Influence ML model (model poisoning)

Compromised toolchain: adversary inside training pipeline

Diagram showing the toolchain with data owners, Libs, Trainer, ML model, Prediction Service Provider API, and a Client. The diagram highlights the flow of data and the potential for privacy violations.


Malicious prediction service

Data owners → Dataset → Trainer → ML model → Prediction Service Provider API → Database

Profile users

Add: “X uses app”

Is this app malicious?

Compromised input

Malicious client

Invert model, infer membership

Malicious client

Data owners

Dataset

Trainer

Libs

ML model

Prediction Service Provider API

Client

Extract/steal model

Oblivious Neural Network Predictions via MiniONN Transformations

N. Asokan,
https://asokan.org/asokan/
@nasokan

(Joint work with Jian Liu, Mika Juuti, Yao Lu)
Machine learning as a service (MLaaS)

violation of clients’ privacy
Running predictions on client-side

model theft
evasion
model inversion
Oblivious Neural Networks (ONN)

Given a neural network, is it possible to make it oblivious?

• server learns nothing about clients' input;

• clients learn nothing about the model.
Example: CryptoNets

- High throughput for batch queries from same client
- High overhead for single queries: 297.5s and 372MB (MNIST dataset)
- Cannot support: high-degree polynomials, comparisons, …
MiniONN: Overview

Blinded input ➔ oblivious protocols ➔ Blinded predictions

- Low overhead: ~1s
- Support all common neural networks
Example \( z = W' \cdot f(W \cdot x + b) + b' \)

\[
\begin{align*}
    x &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix},
    W &= \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix},
    b &= \begin{bmatrix} b_1 \\ b_2 \end{bmatrix},
    W' &= \begin{bmatrix} w'_{1,1} & w'_{1,2} \\ w'_{2,1} & w'_{2,2} \end{bmatrix},
    b' &= \begin{bmatrix} b'_1 \\ b'_2 \end{bmatrix}
\end{align*}
\]

All operations are in a finite field \( \mathbb{Z}_N \)
Core idea: use secret sharing for oblivious computation

Use efficient cryptographic primitives (2PC, additively homomorphic encryption)

$W' \cdot [] + b'$

$W \cdot [] + b$

$y^c + y^s = y'$

$x^c + x^s = x'$

(y$^c + y^s = y$)

(x$^c + x^s = x$)
Secret sharing initial input $x$

$x_1^c, x_2^c \leftarrow \$ Z_N$

$x_1^s := x_1 - x_1^c, \quad x_2^s := x_2 - x_2^c$

Note that $x^c$ is independent of $x$. Can be pre-chosen
Oblivious linear transformation $\mathbf{W} \cdot \mathbf{x} + \mathbf{b}$

\[
\begin{bmatrix}
    w_{1,1} & w_{1,2} \\
    w_{2,1} & w_{2,2}
\end{bmatrix}
\begin{bmatrix}
    x_1 \\
    x_2
\end{bmatrix}
+ 
\begin{bmatrix}
    b_1 \\
    b_2
\end{bmatrix}
= 
\begin{bmatrix}
    w_{1,1} & w_{1,2} \\
    w_{2,1} & w_{2,2}
\end{bmatrix}
\begin{bmatrix}
    x_1^s + x_1^c \\
    x_2^s + x_2^c
\end{bmatrix}
+ 
\begin{bmatrix}
    b_1 \\
    b_2
\end{bmatrix}
\]

Compute locally by the server

Dot-product
Oblivious linear transformation: dot-product

\[ r_{1,1}, r_{1,2}, r_{2,1}, r_{2,2} \overset{\$}{\leftarrow} \mathbb{Z}_N \]
\[ c_{1,1} = E(w_{1,1}x^c - r_{1,1}) \]
\[ c_{1,2} = E(w_{1,2}x^c - r_{1,2}) \]
\[ c_{2,1} = E(w_{2,1}x^c - r_{2,1}) \]
\[ c_{2,2} = E(w_{2,2}x^c - r_{2,2}) \]
\[ \mathbf{v}_1 = r_{1,1} + r_{1,2} \]
\[ \mathbf{u}_1 = w_{1,1}x^c + w_{1,2}x^c - (r_{1,2} + r_{1,1}) \]
\[ \mathbf{v}_2 = r_{2,1} + r_{2,2} \]
\[ \mathbf{u}_2 = w_{2,1}x^c + w_{2,2}x^c - (r_{2,1} + r_{2,2}) \]

\[ \mathbf{u} + \mathbf{v} = W \cdot \mathbf{x}^c; \text{ Note: } \mathbf{u}, \mathbf{v}, \text{ and } W \cdot \mathbf{x}^c \text{ are independent of } \mathbf{x}. \]

\[ <\mathbf{u}, \mathbf{v}, \mathbf{x}^c> \text{ generated/stored in a precomputation phase} \]
Oblivious linear transformation $W \cdot x + b$

\[
\begin{bmatrix}
  w_{1,1} & w_{1,2} \\
  w_{2,1} & w_{2,2}
\end{bmatrix}
\begin{bmatrix}
  x_1 \\
  x_2
\end{bmatrix}
+ 
\begin{bmatrix}
  b_1 \\
  b_2
\end{bmatrix}
= 
\begin{bmatrix}
  w_{1,1} & w_{1,2} \\
  w_{2,1} & w_{2,2}
\end{bmatrix}
\begin{bmatrix}
  x_1^s + x_1^c \\
  x_2^s + x_2^c
\end{bmatrix}
+ 
\begin{bmatrix}
  b_1 \\
  b_2
\end{bmatrix}
\]

\[
\begin{bmatrix}
  w_{1,1}(x_1^s + x_1^c) + w_{1,2}(x_2^s + x_2^c) + b_1 \\
  w_{2,1}(x_1^s + x_1^c) + w_{2,2}(x_2^s + x_2^c) + b_2
\end{bmatrix}
= 
\begin{bmatrix}
  w_{1,1}x_1^s + w_{1,2}x_2^s + b_1 + w_{1,1}x_1^c + w_{1,2}x_2^c \\
  w_{2,1}x_1^s + w_{2,2}x_2^s + b_2 + w_{2,1}x_1^c + w_{2,2}x_2^c
\end{bmatrix}
\]

\[
\begin{bmatrix}
  w_{1,1}x_1^s + w_{1,2}x_2^s + b_1 + u_1 \\
  w_{2,1}x_1^s + w_{2,2}x_2^s + b_2 + u_2
\end{bmatrix}
+ 
\begin{bmatrix}
  v_1 \\
  v_2
\end{bmatrix}
\]

12
Oblivious linear transformation $W \cdot x + b$

$$
\begin{bmatrix}
  w_{1,1} & w_{1,2} \\
  w_{2,1} & w_{2,2}
\end{bmatrix}
\begin{bmatrix}
  x_1 \\
  x_2
\end{bmatrix}
+ 
\begin{bmatrix}
  b_1 \\
  b_2
\end{bmatrix}
=
\begin{bmatrix}
  w_{1,1} & w_{1,2} \\
  w_{2,1} & w_{2,2}
\end{bmatrix}
\begin{bmatrix}
  x_1^s + x_1^c \\
  x_2^s + x_2^c
\end{bmatrix}
+ 
\begin{bmatrix}
  b_1 \\
  b_2
\end{bmatrix}
$$

$$
\begin{bmatrix}
  w_{1,1}(x_1^s + x_1^c) + w_{1,2}(x_2^s + x_2^c) + b_1 \\
  w_{2,1}(x_1^s + x_1^c) + w_{2,2}(x_2^s + x_2^c) + b_2
\end{bmatrix}
=
\begin{bmatrix}
  w_{1,1}x_1^s + w_{1,2}x_2^s + b_1 + w_{1,1}x_1^c + w_{1,2}x_2^c \\
  w_{2,1}x_1^s + w_{2,2}x_2^s + b_2 + w_{2,1}x_1^c + w_{2,2}x_2^c
\end{bmatrix}
$$

$$
\begin{bmatrix}
  w_{1,1}x_1^s + w_{1,2}x_2^s + b_1 + u_1 \\
  w_{2,1}x_1^s + w_{2,2}x_2^s + b_2 + u_2
\end{bmatrix}
+ 
\begin{bmatrix}
  v_1 \\
  v_2
\end{bmatrix}
:=
\begin{bmatrix}
  y_1^s \\
  y_2^s
\end{bmatrix}
+ 
\begin{bmatrix}
  y_1^c \\
  y_2^c
\end{bmatrix}
$$
Oblivious activation/pooling functions $f(y)$

Piecewise linear functions e.g.,

- ReLU: $x := \max(y, 0)$
- Oblivious ReLU: $x^s + x^c := \max(y^s + y^c, 0)$

- easily computed obliviously by a garbled circuit
Oblivious activation/pooling functions $f(y)$

Smooth functions e.g.,

- **Sigmoid**: $x := 1/(1 + e^{-y})$
- **Oblivious sigmoid**: $x^s + x^c := 1/(1 + e^{-(y^s + y^c)})$
  - approximate by a piecewise linear function
  - then compute obliviously by a garbled circuit
  - empirically: $\sim 14$ segments sufficient
Combining the final result

They can jointly calculate max(y₁, y₂) (for minimizing information leakage)

\[ y_1 := y_1^s + y_1^c \]
\[ y_2 := y_2^s + y_2^c \]
Core idea: use secret sharing for oblivious computation

\[ W' \cdot [] + b' \]

\[ x'^c \]

\[ y'^c \]

\[ f([]) \]

\[ x'^s \]

\[ y'^s \]

\[ x'^c + x'^s = x' \]

\[ (y'^c + y'^s = y') \]

\[ x^c + x^s = x \]

\[ y^c + y^s = y \]
## Performance (for single queries)

<table>
<thead>
<tr>
<th>Model</th>
<th>Latency (s)</th>
<th>Msg sizes (MB)</th>
<th>Loss of accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST/Square</td>
<td>0.4 (+ 0.88)</td>
<td>44 (+ 3.6)</td>
<td>none</td>
</tr>
<tr>
<td>CIFAR-10/ReLU</td>
<td>472 (+ 72)</td>
<td>6226 (+ 3046)</td>
<td>none</td>
</tr>
<tr>
<td>PTB/Sigmoid</td>
<td>4.39 (+ 13.9)</td>
<td>474 (+ 86.7)</td>
<td>Less than 0.5% (cross-entropy loss)</td>
</tr>
</tbody>
</table>

Pre-computation phase timings in parentheses

PTB = Penn Treebank
MiniONN pros and cons

300-700x faster than CryptoNets

Can transform any given neural network to its oblivious variant

Still ~1000x slower than without privacy

Server can no longer filter requests or do sophisticated metering

Assumes online connectivity to server

Reveals structure (but not params) of NN
Can trusted computing help?

Hardware support for
- Isolated execution: Trusted Execution Environment
- Protected storage: Sealing
- Ability to report status to a remote verifier: Attestation

Other Software
Trusted Software
Protected Storage
Root of Trust

Cryptocards: https://www.ibm.com/security/cryptocards/
Trusted Platform Modules: https://www.infineon.com/tpm/
ARM TrustZone: https://www.arm.com/products/security-on-arm/trustzone
Intel Software Guard Extensions: https://software.intel.com/en-us/sgx
Using a client-side TEE to vet input

1. Attest client’s TEE app
2. Provision filtering policy
3. Input
4. Input, “Input/Metering Certificate”
5. MiniONN protocol + “Input/Metering Certificate”

MiniONN + policy filtering + advanced metering
Using a client-side TEE to run the model

3. Input

1. Attest client’s TEE app

2. Provision model configuration, filtering policy

4. Predictions + “Metering Certificate”

5. “Metering Certificate”

MiniONN + policy filtering + advanced metering
+ disconnected operation + performance + better privacy
- harder to reason about model secrecy
Using a server-side TEE to run the model

1. Attest server’s TEE app
2. Input
3. Provision model configuration, filtering policy
4. Prediction

MiniONN + policy filtering + advanced metering
- disconnected operation + performance + better privacy
MiniONN: Efficiently transform any given neural network into oblivious form with no/negligible accuracy loss

Trusted Computing can help realize improved security and privacy for ML

ML is very fragile in adversarial settings

https://eprint.iacr.org/2017/452
ACM CCS 2017
Research collaboration with top academic groups

Funding is good, but **active research collaboration** is more valuable
- real problem insights, access to data & technology, prospects for tech transfer

Subcontracted work will not fly
- aim for publishable research, **partnership** (not management)

“Open IP” is mutually beneficial
- Case example: Intel Collaborative Research Institute ([http://www.icri-sc.org/](http://www.icri-sc.org/))