The Opening Gambit
The Study
The Proposal
Abstract
We present a new general technique for protecting clients in distributed systems against Remote Man-in-the-Middle (R-MATE) attacks. Such attacks occur in settings where an adversary has physical access to an untrusted client device and can exploit the adversary’s ability to tamper with the client device itself. The adversary’s goal is to prevent attacks from being detected by the system. In our system, the trusted server continually monitors the adversary’s attack, making exploitation difficult without being detected by the server.

1. Introduction
Remote Man-in-the-Middle (R-MATE) attacks occur in settings where an adversary has physical access to a client and can exploit the adversary’s ability to tamper with the client device itself. In our system, the trusted server continually monitors the adversary’s attack, making exploitation difficult without being detected by the server.

1.1 Overview
In each of the scenarios above the adversary’s goal is to tamper with the client code and data under his control. The trusted server’s goal is to detect any such integrity violations, after which countermeasures (such as suspending connections, terminating sessions, etc.) can be launched.

Security mechanisms. In this paper we present a system that achieves protection against R-MATE attacks through the use of code diversity and continuous code replacement. In our system, the trusted server continually monitors the adversary’s attack, making exploitation difficult without being detected by the server.

Limitations. Our system specifically targets distributed applications which have frequent client-server communication, since client tampering can only be detected at client-server interaction events. Furthermore, since we use code diversity to detect an attack, it cannot completely prevent it. One goal is therefore to reduce the detection time of attacks, which should lead to completely prevent any tampering of client code. For even shorter attack times, we must consider targets for our system. To see this, consider the following timeline in the history of a distributed application running under our system:

- $s_1$: Client tampers
- $s_2$: Server detects
- $t_1$: Client responds

The $s_i$ are interaction events, points in time when clients communicate with servers either to exchange application data or to perform code updates. At time $s_1$, the client tampers with the code. The attack is detected by the server at time $s_2$. At time $t_1$, after an interval $t_1$, the client requests the update, and the server detects the attack. At time $t_2$, after an interval $t_2$, the client requests the update, and the server detects the attack. At time $t_3$, finally, the server issues a response, perhaps by shutting...
To: authors@cs.ux.edu

Hi again! We think our system can break yours! Can you please send us your code? 😑
\( \phi \) type operator =
\[
| NOP \\
| LC of operand * value * binop \\
| MR of operand * value * operand * binop \\
| MW of operand * value * operand * binop \\
| MV of operand * operand \\
\]

\( f: \mathbb{N} \to \mathbb{N} \) typecheck.
I have few recollections of the work. I looked at the tech report, and it was like seeing a new paper for the first time.

1) What is $\varphi$?  
2) Where should $f$ be used?  
3) Fix type mismatch?
[T]he 2005 NSF Grant Policy Manual ... states that

c) Investigators and grantees are **encouraged to share software** and inventions created under the grant or otherwise make them or their products widely available and usable.
From: legal@cs.ux.edu

... to the extent such records may exist, they will not be produced pursuant to ORA.
As you are aware, the ORA states that ... Hence, information about this system ... is public record and subject to disclosure. ... I must therefore reiterate my request for emails between the authors related to the system to be released.
We will also make our data and software available to the research community when appropriate.
Why?
So, why am I telling you this story? Well, what we learned from this story, is that people are not always willing to share.

There are 3 reasons for why, as a researcher, you would want to share the artifacts that your research produces with other researchers.

First of all, you want the reviewers and readers of your work to trust it. In other words, it should be possible to repeat your experiments and get the same results. And if they don’t have access to your code and data, how can they? That’s what we call repeatability.

Second, it should be possible to independently verify the claims in your paper. That’s what we call reproducibility. And, since in computer systems, we compress 100,000 lines of code into 15 pages of a research paper, completely understanding a work may require access to the code and data that went into it.

Finally, like your mother told you and your siblings, it is good to share. In science, it’s good to share because it allows others to build on your work, and that will advance the field. And sharing for the good of scientific progress, we call benefaction.
Repeatability

System

Experiments
- Methodology
- Environment
- Benchmarks

Properties

Publication

Verify
[T]he ability to re-run the exact same experiment with the same method on the same or similar system and obtain the same or very similar result.

Vitek, Kalibera: R3 – Repeatability, Reproducibility and Rigor
Reproducibility

What are the Claims?

System
- Experiments
  - Methodology
  - Environment
  - Benchmarks

New System

Different Experiments
- Methodology
- Environment
- Benchmarks

Verify Claims

Properties

Publication

What are the Claims?
Reproducibility

[The] independent confirmation of a scientific hypothesis through reproduction by an independent researcher/lab...

[Is] carried out after a publication, based on the information in the paper...

Vitek, Kalibera: R3 – Repeatability, Reproducibility and Rigor
Benefaction

System

Experiments
- Methodology
- Environment
- Benchmarks

Properties

Publication

New System

New Properties
The avoidance of needless replication of work in order to better advance scientific progress.
The Study
Weak Repeatability

Do authors make the source code used to create the results in their article available, and will it build?
* Preliminary results - more about this later!
Reasons for not Sharing?

The email responses we received were pleasant, accommodating, and apologetic if code could not be provided.
The good news ... I was able to find some code. I am just hoping that it ... matches the implementation we ... used for the paper.

git tag -a pldi-final

Versioning
Unfortunately the current system is not mature ... We are actively working on a number of extensions ...
The code was **never intended to be released** so is not in any shape for general use.

Publishable $\neq$ Sharable?

No Inention to Share
[Our] prototype ... included many moving pieces that only **student** knew how to operate ... **he** left.
... the server in which my implementation was stored had a **disk crash** ...

... three disks crashed ...

Sorry for that.
[Therefore] we will not provide the source code outside the group.
Ultimately the product groups sponsor our employment. We are very sorry that we can't share the code ...
Unfortunately, the system sources are not meant to be open source (the code is partially property of three universities).
and we also have a **software license agreement** that the University would need to sign.
... few people would manage to get it to work on new hardware.
... based on earlier (bad) experience, we [want] to make sure that our implementation is not used in situations that it was not meant for.
… we have an agreement with the utility company, and we cannot release the code because of the potential privacy risks …
The code ... is complete, but **hardly usable by anyone** other than the authors ... due to our decision to use Template Haskell ...
The Proposal
How Do We Fix This?

- **System**
  - Environment
  - Benchmarks

- **Virtual Machine**

  - performance?
  - security?
  - longevity?

- if you build it, they still won’t come...
How Do We Fix This?

This measure was put in place to reassure authors who felt this would be too radical a change to the process of evaluating conference paper submissions.

- 55 accepted papers
- 20 (36%) artifacts submitted for review
- 12 (22%) met or exceeded expectations
[As all spheres of human activity are affected by the interplay between social structure and individual agency, sociology has gradually expanded its focus to further subjects, such as …, and the role of social activity in the development of scientific knowledge. http://en.wikipedia.org/wiki/Sociology].
Our study found papers with:

- no author email addresses
- wrong email addresses
- wrong links to code
- overloaded project names
- artifact (code, data, media, docs, …)
- availability (no access, access)
- expense (free, non-free)
- distribution form (source, binary, service)
- expiration date
- comment

http://reproducibility.cs.arizona.edu;
mailto:collberg@gmail.com;

code: access, free, source;
data: access, free, source, “sanitized”;
support: L1, free, 2015-12-31;
Epilogue
What Happened Next?
Like you, I have been on the receiving end of stonewalling, silence, and maybe even lies when trying to get hold of code from others. Your emails ... completely resonated with me.

* IRB - Institutional Review Board
Examining “Reproducibility in Computer Science”

1 What We Are Doing

Welcome to repo-repe-repro: the repository to repeat an experiment in “reproducibility”!

A group led by Christian Collberg attempted to evaluate the buildability of artifacts from research papers. Our goal is to allow the community to review and reconstruct their findings. Note: We are not the original authors! If you have questions about the original study, please contact them, not us!

We are grateful to Collberg, et al. for initiating this discussion and making all their data available. This is a valuable service based on an enormous amount of manual labor. Even if we end up disagreeing with their findings, we remain deeply appreciative of their service to the community by highlighting these important issues.

We do disagree with Collberg, et al.’s use of the term “reproducibility”. Many people, including ourselves, associate it with an independent reconstruction of a work. This paper, for instance, spells out the difference between repeatability and reproducibility and provides interesting examples.
Practical?

Code?

Builds?

Weakly Repeatable

1. Reexamine failed builds
2. Request author feedback
Conclusions
1. Opening Gambit
2. Study
3. Proposal
1. Demanding everyone to share code always is unrealistic.

2. Sharing specifications are a low-cost alternative that can be implemented now.

3. We believe sharing specifications will be an incentive to authors to produce solid computational artifacts.
Questions?