EVMPatch: Timely Patching of Ethereum Smart Contracts with EVM Bytecode Rewriting

Michael Rodler, Wenting Li,
Ghassan Karame, Lucas Davi

University of Duisburg-Essen, NEC Labs Europe

How to protect smart contracts after deployment?

Why don’t you deploy a patch?

Ethereum Smart Contracts are **immutable**

Patching Smart Contracts

- A lot of prior work on vulnerability detection
  - Symbolic execution (e.g., Oyente, teEther, EthBMC, …)
  - Static analysis (e.g., Securify, eThor, …)
  - Dynamic analysis (e.g., Sereum, TXSpector, …)
  - ...
- We regularly observe incidents on the blockchain.
- **We need to enable smart contract developers to patch new issues!**

Existing Patching Strategies

Migration to a New Contract

Deprecate old contract, deploy new contract, **manually migrate** state to new contract.

Upgradable Contract using a Proxy Contract

Contract is split into two:

- proxy contract
- logic contract

Requires **manual conversion**;

must ensure **storage layout compatibility**

Are Upgradable Contracts Practical?

Study with 6 Developers
(4 with "production-grade" smart contract experience)

Task	Median Minutes	Median Reported Confidence (1-7)
Manual Patching	47.5	6
Manual Upgradable Contract (Proxy Pattern)	*62.5*	2.5

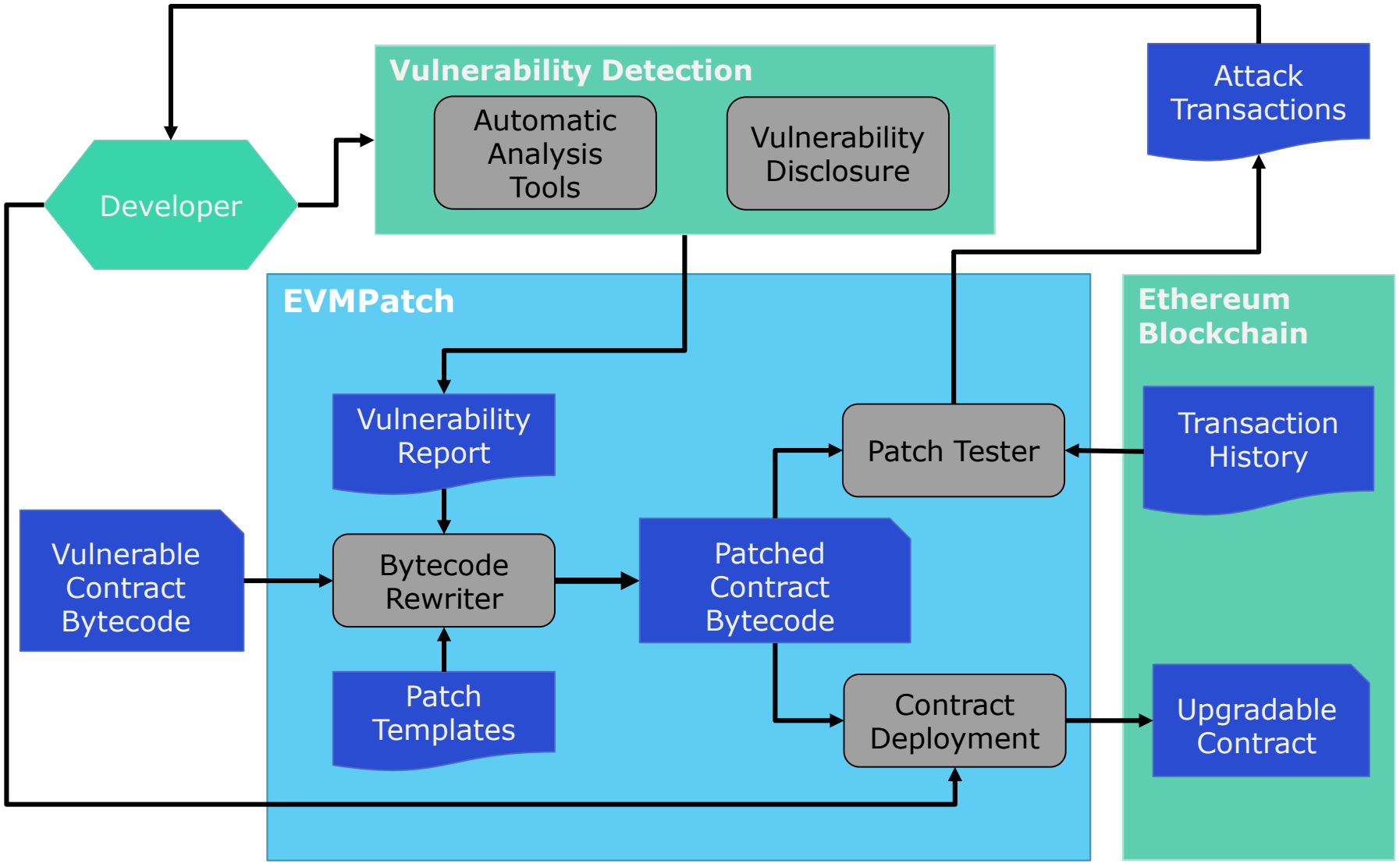
None of the manually created upgradable contracts were fully functional!

Upgrading smart contracts is
cumbersome,
time-consuming,
and **error-prone.**

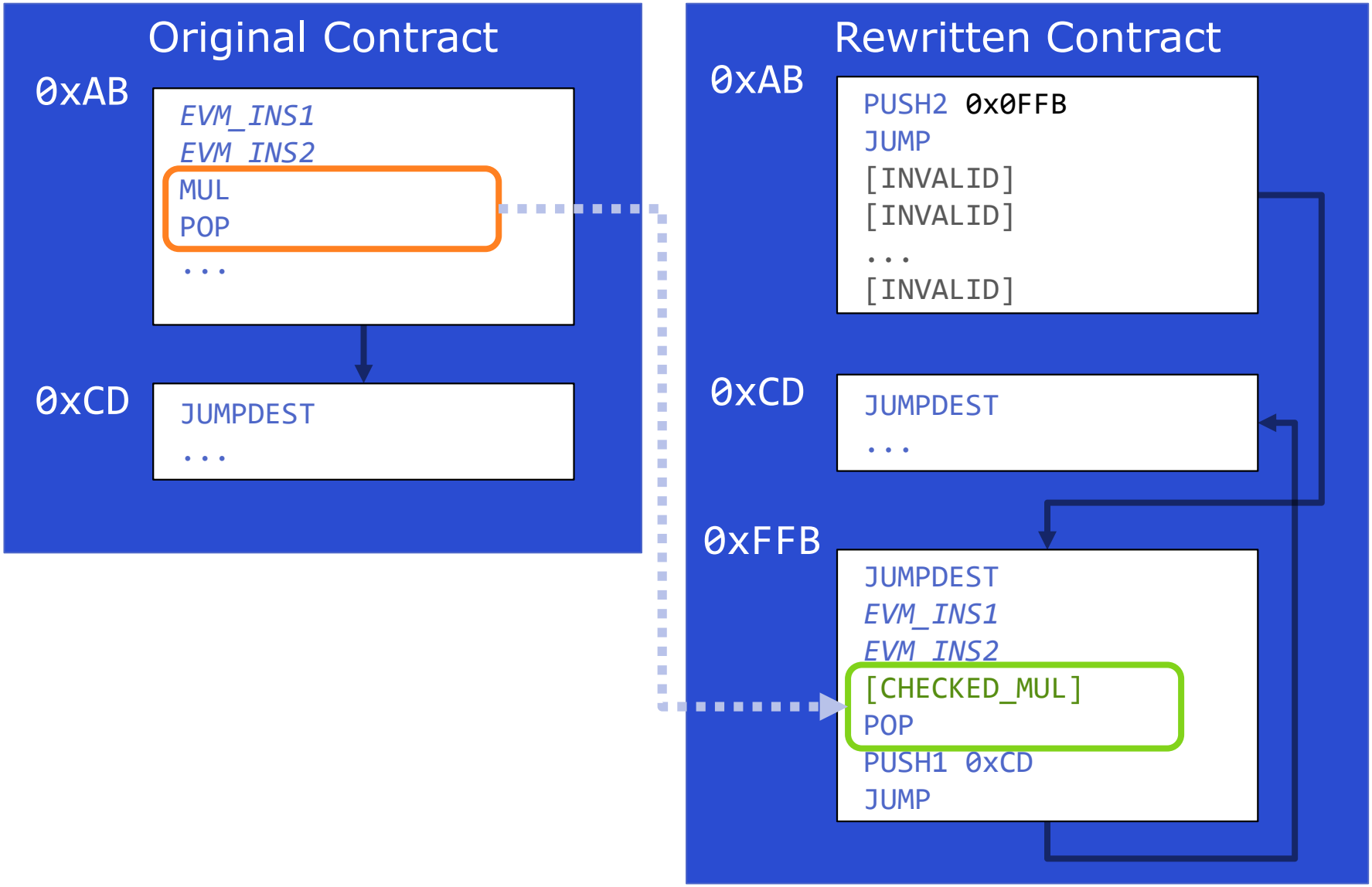
Introducing EVMPatch

- Fully **automated patching** framework
- Automates the delegatecall-**proxy pattern**
  - Automatic conversion to proxy pattern
  - Deployment of contract and upgrades
- Patching with **bytecode rewriting**
  - Template-based patching with custom DSL
  - Naturally preserves storage-layout
- **Differential patch testing**
  - Ensure equivalent behavior:
    original vs patched contract

EVMPatch Architecture



EVMPatch Integer Overflow Check



Evaluation Results: Attacks

Evaluation on 5 known exploited ERC-20 Token Contracts

Contract	CVE	# Transactions	# Attacks
BEC	2018-10299	424 229	1
SMT	2018-10376	56 555	1
UET	2018-10468	24 034	55
SCA	2018-10706	292	1
HXG	2018-11239	1497	9

Comparison with manual patches (SafeMath)

EVMPatch'ed contracts…

- Prevent same attacks as SafeMath
- Same behavior as original on non-attacks
- Comparable overhead to source-level patches

Evaluation Results: Practicality

- Additional Cost due to Gas Overhead
  - Per Transaction: < 0.01$
  - Per Upgrade: < 0.20$
- Developer Study

Task	Median Minutes	Median Reported Confidence (1-7)
EVMPatch Patch+Deploy	1.5	-
New EVMPatch Template	4.0	7

About **5 minutes** to **patch** and **deploy** a **new** type of **vulnerability** with EVMPatch!

EVMPatch: Timely Patching of Ethereum Smart Contracts with EVM Bytecode Rewriting

Michael Rodler, Wenting Li, Ghassan Karame, Lucas Davi

- **Practical Post-Deployment Protection**
- **Efficient EVM Bytecode Patching**
- **Timely Patching of Vulnerabilities**
- **Automated Upgradable Contracts**

michael.rodler@uni-due.de

https://udue.de/evmpatch

UNIVERSITÄT DUISBURG ESSEN

Open-Minded

NEC