NPTEL ONLINE CERTIFICATION COURSES

Blockchain and its applications

Bishakh Chandra Ghosh

Department of Computer Science & Engineering
Indian Institute of Technology Kharagpur

Lecture 51: Hyperledger Aries

CONCEPTS COVERED

- Aries Overview
- Aries Architecture
- Installation and usage

KEYWORDS

- **Digital Credentials**
- **Aries**

Hyperledger Aries

Hyperledger Aries provides a **shared, reusable, interoperable tool** kit designed for initiatives and solutions focused on creating, transmitting and storing verifiable **digital credentials**.

It is infrastructure for **blockchain-rooted**, peer-to-peer interactions.

Aries agent frameworks -.

- **Aries Cloud Agent - Python (ACA-Py)**
  - For any non-mobile application. Has production deployments.
- Aries Framework - .NET
- Aries Static Agent - Python

https://www.hyperledger.org/use/aries

Aries Architecture

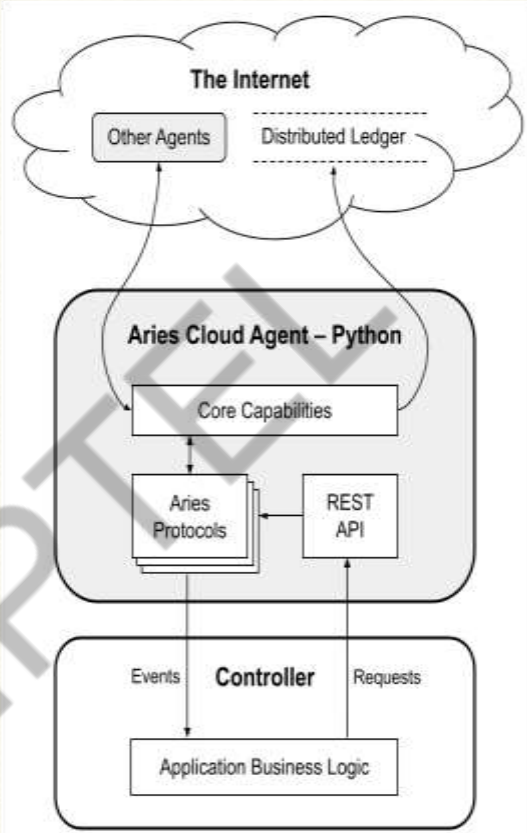
ACA-Py

- ACA-Py exposes a REST API to access Aries capabilities

- Write a '**Controller**' to implement application business logic.

- Controller sends **HTTP requests** (REST) to execute commands.

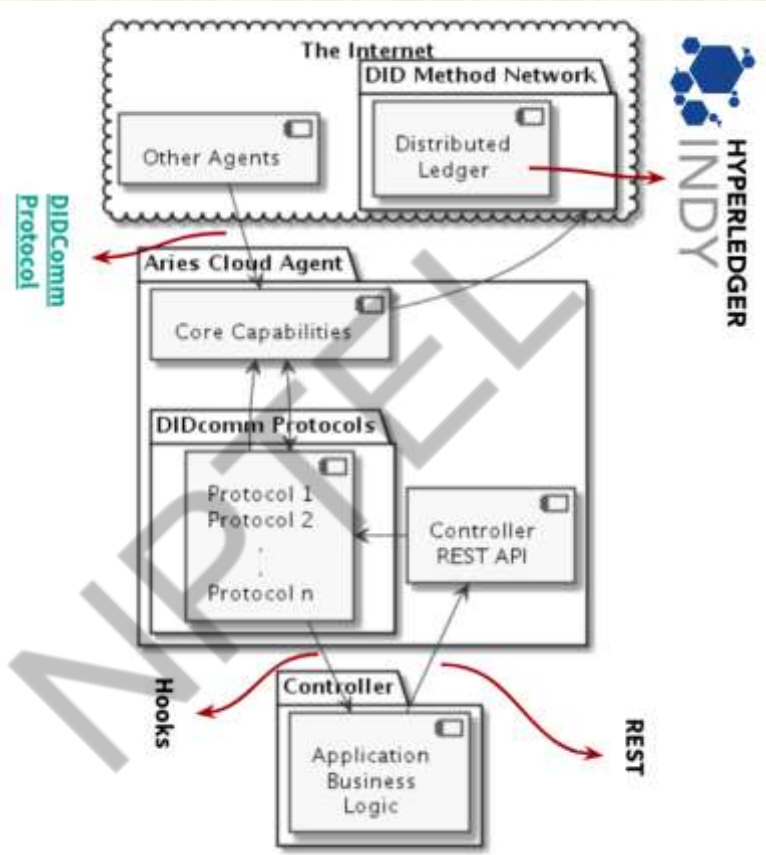
Response events are fed back to the controller as **webhooks**.



Aries Architecture

The agent -

- Configured via command line parameters
- Interacts with other agents via pluggable transports
- Manages storage, ledger with pluggable implementations
- Manages messages and protocol state
- Invokes protocols (configurable set)
- Driven by a controller
    - Sends events to controller
    - Exposes an HTTP JSON administrative API to controller



Installation

- Install libindy and indy-cli
  - `apt-key adv --keyserver keyserver.ubuntu.com --recv-keys CE7709D068DB5E88`
  - `apt-add-repository "deb https://repo.sovrin.org/sdk/deb bionic master" -y`
  - `apt-get update`
  - `apt-get install -y libindy indy-cli`

- Install python3-indy
  - `python3 -m pip install python3-indy`

- Install aries-cloudagent
  - `python3 -m pip install aries-cloudagent`

Installation

If you installed the PyPi package, the executable aca-py should be available on your PATH.

Use the following commands to check the version, list the available modes of operation, and see all of the command line parameters:

- `aca-py --version`
- `aca-py --help`
- `aca-py provision --help`
- `aca-py start --help`

Starting Aries Agent

Provisioning a Wallet

It is possible to provision an Indy wallet before running an agent:

```
aca-py provision --wallet-type indy --seed $SEED
```

Use the SEED to configure existing nodes in the indy pool, e.g. Stewards.

Starting Aries Agent

When starting an agent instance, at least one ***inbound*** and one ***outbound*** **transport** MUST be specified.

For example:
aca-py start --inbound-transport http 0.0.0.0 8000 --outbound-transport http

More Parameters:

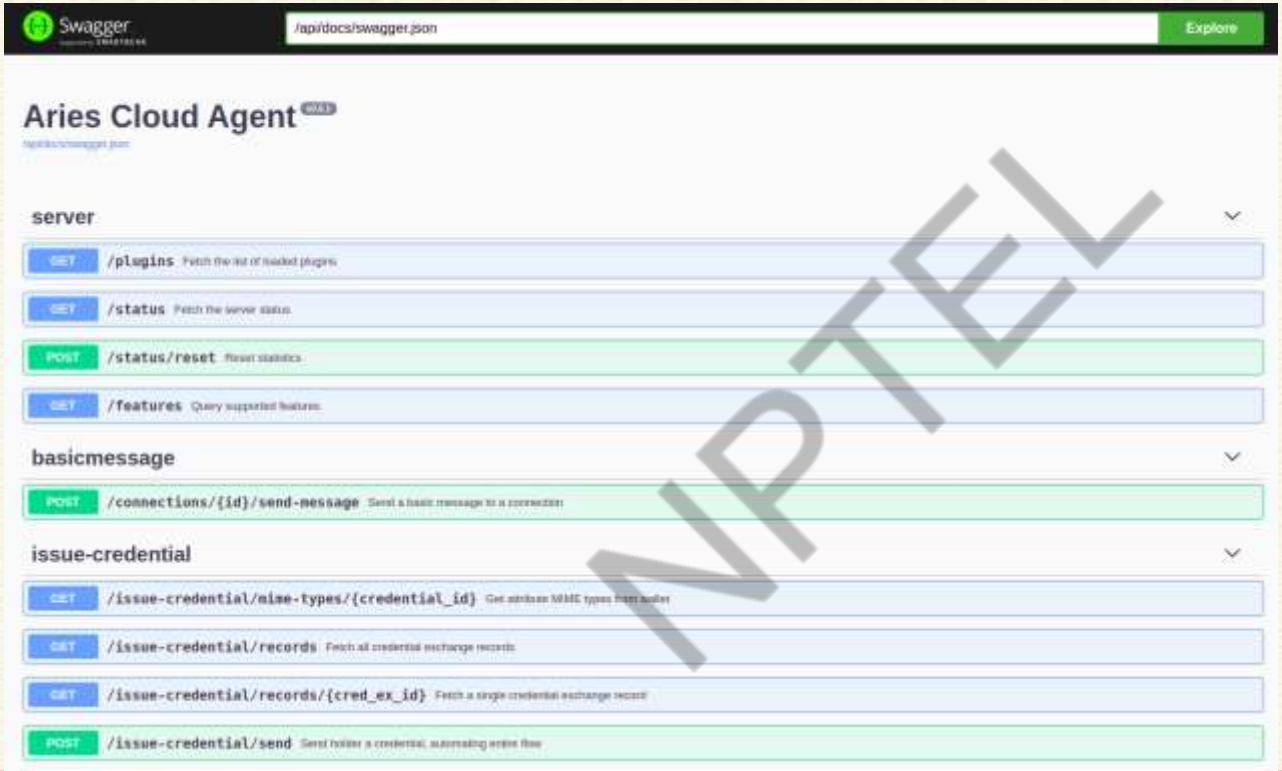
- **Transports**: inbound, outbound, endpoint
- **Logging/debugging settings**
- **Label**: self-attested agent name
- **Wallet** implementation and related info
- For controller: **Admin API** configuration
  - URL
  - Security selection
- Protocol **automation flags**
- **Ledger parameters** (e.g. genesis URL, etc.)
- Add timing information to messaging
- Optional protocols to load
- From controller: **Event webhook URL**

Using Aries with Indy

```
aca-py start --inbound-transport http 0.0.0.0 8000 \
 --outbound-transport http \
 --admin-insecure-mode \
 --admin 0.0.0.0 8001 \
 --seed 000000000000000000000000Steward1 \
 --replace-public-did \
 --wallet-type indy \
 --genesis-file
/home/bishakh/Documents/TA/Blockchain/indytutorial/pool1.txn \
--wallet-name agent1 \
 --wallet-key agent1 \
 --log-level debug \
 --log-file /tmp/arieslog.txt
--admin-insecure-mode \
 --admin 0.0.0.0 8001 \
```

For experimentation and debugging only

Aries Admin OpenAPI Interface



Conclusion

- Aries
- Communication between participants
- Digital Credentials

NPTEL ONLINE CERTIFICATION COURSES

Blockchain and its applications

Prof. Shamik Sural
Department of Computer Science & Engineering
Indian Institute of Technology Kharagpur

Lecture 52: Blockchain Security - I

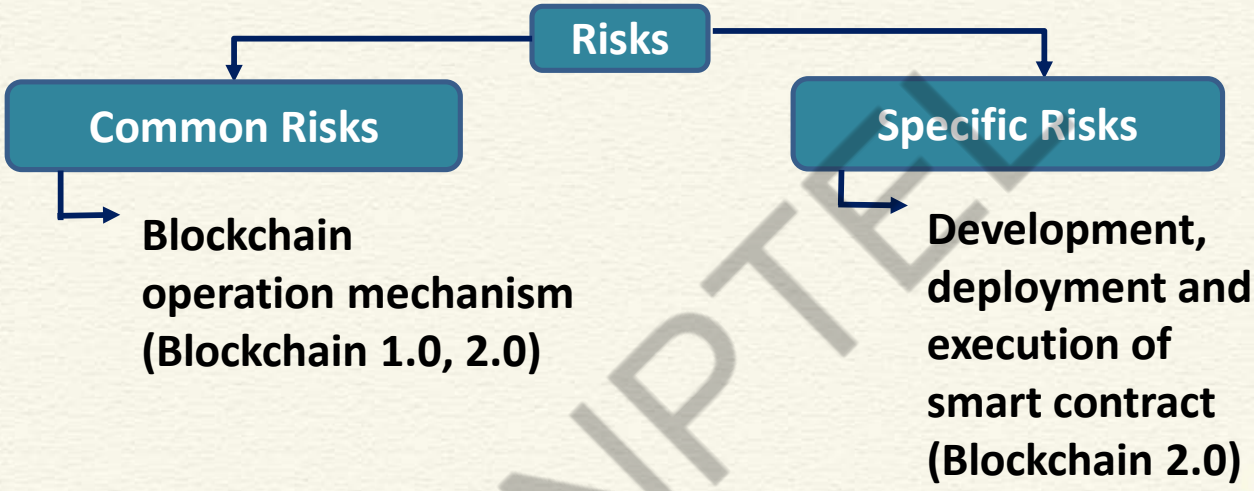
CONCEPTS COVERED

- Risks in Blockchain
- Common Risks and Specific Risks

KEYWORDS

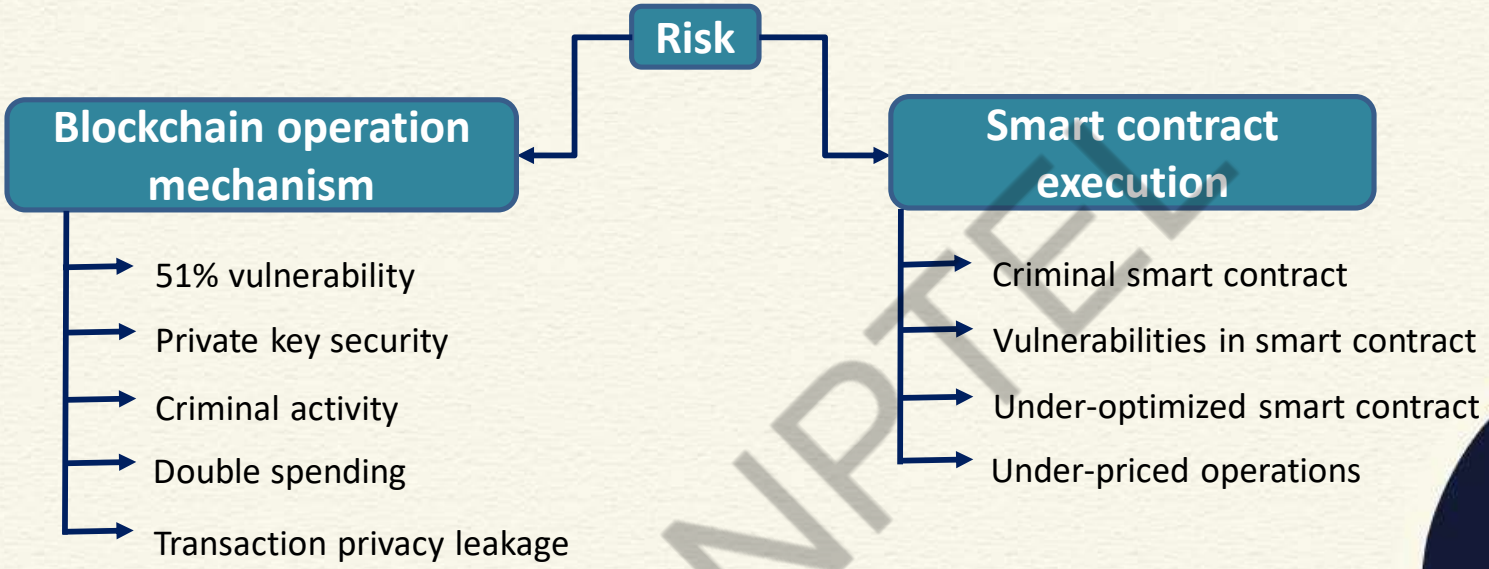
- 51% Vulnerability
- Private Key Security
- Criminal Activities
- Double Spending
- Transaction Privacy

Risks in Blockchain



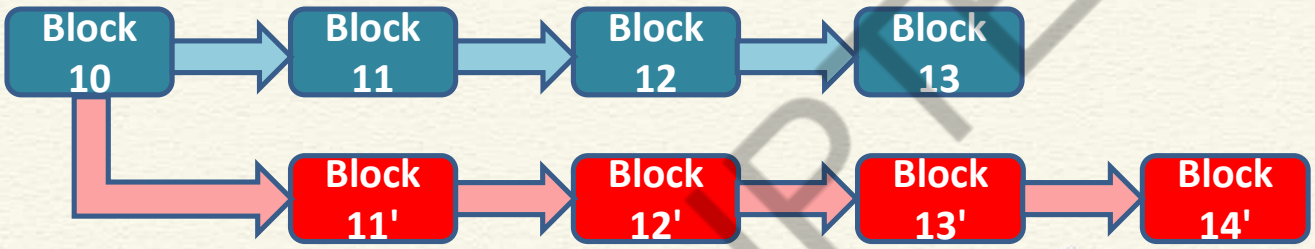
"A Survey on the Security of Blockchain Systems", Xiaoqi Li, Peng Jiang, Ting Chen, Xiapu Luo and Qiaoyan Wen, Future Generation Computer Systems

Risks in Blockchain



Common Risk: 51% Vulnerability

Hashing power



Hashing power

Common Risk: 51% Vulnerability

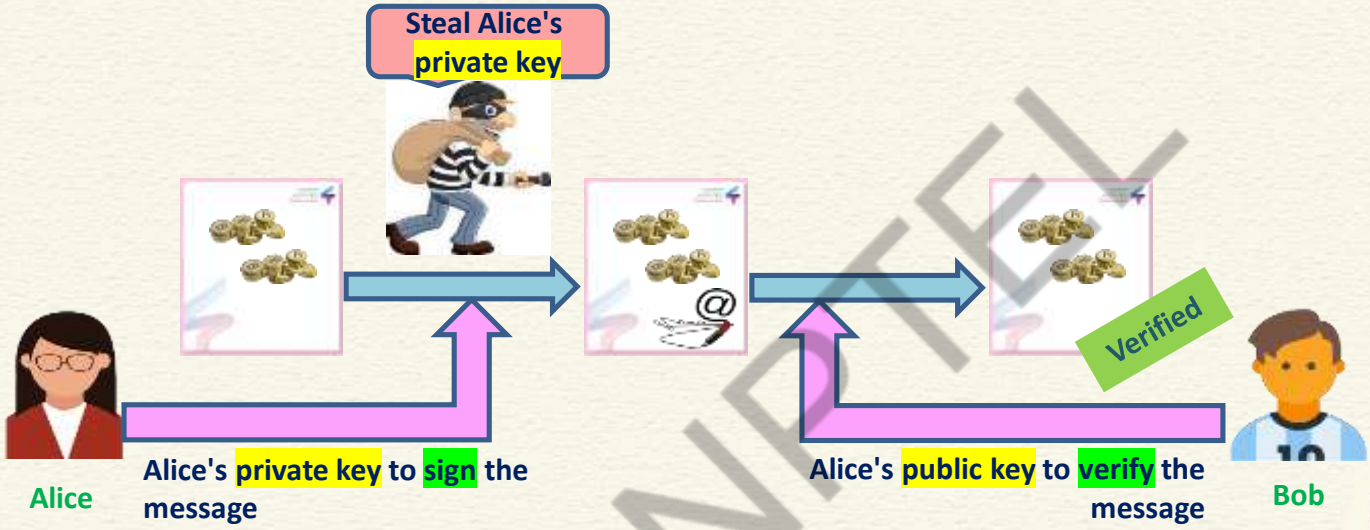
Coins

Block 10 → Block 11 → Block 12 → Block 13

Block 10 → Block 11' → Block 12' → Block 13' → Block 14'

Coins

Common Risk: Private Key Security



Common Risk: Criminal Activities

Ransomware

Underground market

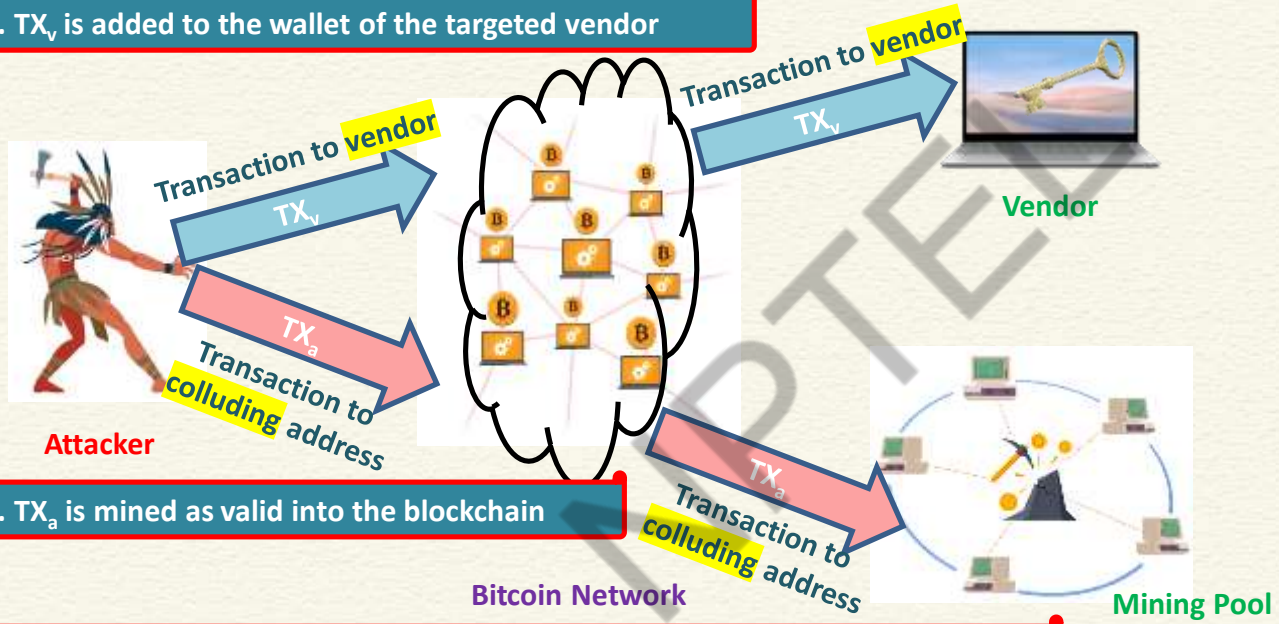
Chaff coin

Money laundering

Valid coin

Common Risk: Double Spending

1. TX_v is added to the wallet of the targeted vendor

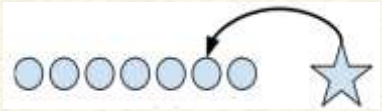


2. TX_a is mined as valid into the blockchain

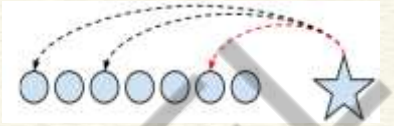
3. The attacker gets TX_v 's output before the vendor detects misbehavior

Common Risk: Transaction Privacy Leakage

A new transaction (the star) which spends an available coin (the second circle from the right)



Transactions and tracing in Bitcoin



Transactions and tracing in Cryptonote

- ✓ Each transaction input explicitly identifies the coin being spent, thus forming a linkage graph
- ✓ Each transaction input identifies a set of coins, including the real coin along with several chaff coins called "mixins."
  - ✓ Many mixins can be ruled out by deduction
  - ✓ The real input is usually the "newest" one

CONCLUSIONS

- Introduced the basic risk types in blockchain
- Discussed in detail some of the common risks

REFERENCES

- **Web resources as mentioned from time to time**

NPTEL ONLINE CERTIFICATION COURSES

Blockchain and its applications

Prof. Shamik Sural
Department of Computer Science & Engineering
Indian Institute of Technology Kharagpur

Lecture 53: Blockchain Security - II

CONCEPTS COVERED

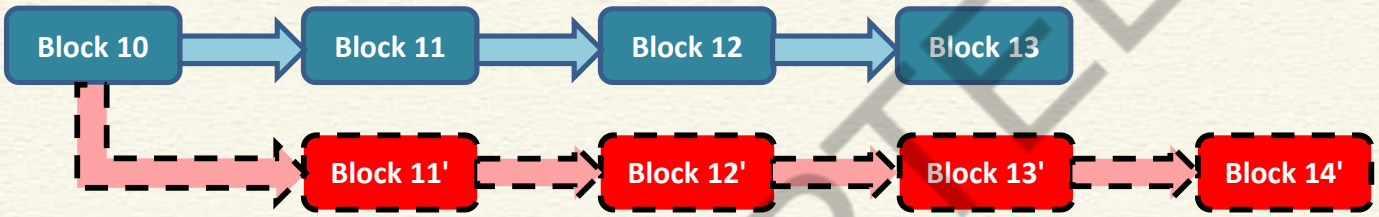
- Selfish Mining Attack
- Different Scenarios and Attacker's Actions

KEYWORDS

- Selfish Mining
- Attacker's Pool
- Public Chain
- Block Suppression

Selfish Mining Attack

Others

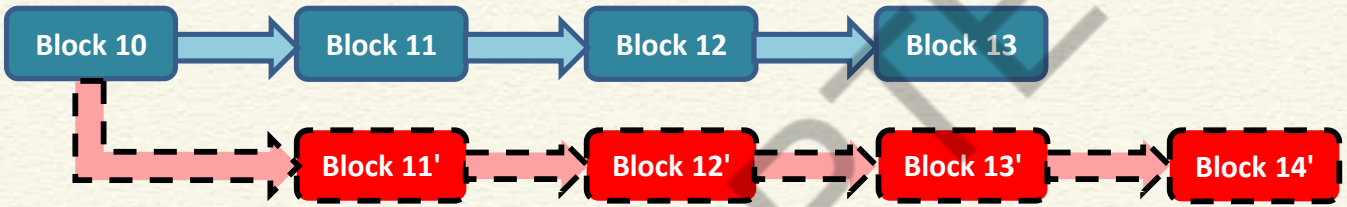


Mining Pool

"Majority Is Not Enough: Bitcoin Mining Is Vulnerable", Ittay Eyal and Emin Guen Sirer, Financial Cryptography, 2014

Selfish Mining Attack

Others



Mining Pool

Selfish Mining Attack

Pool intentionally forking the chain for keeping discovered blocks private

The honest nodes continue to mine on the public chain
The pool mines on its own private branch

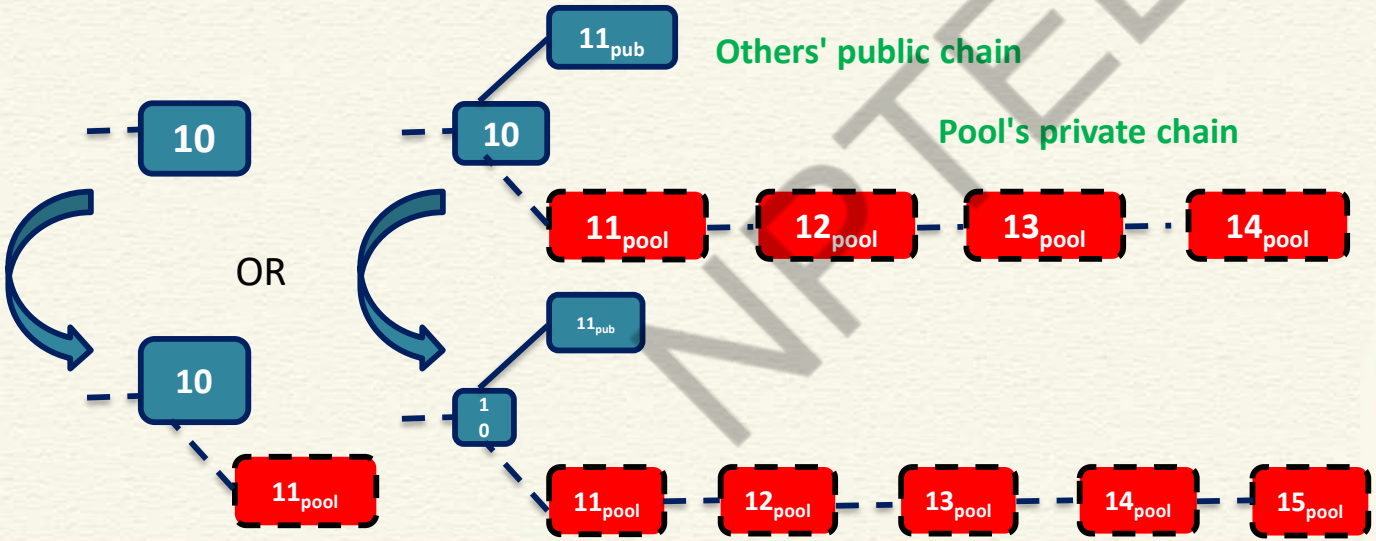
Discovering more blocks by pool develops a longer lead on the public chain, and continues to keep these new blocks private

When the public branch approaches the pool's private branch in length, the selfish miners reveal blocks from their private chain to the public

Selfish Mining Attack

1. Any state but two branches of length 1, pools finds a block

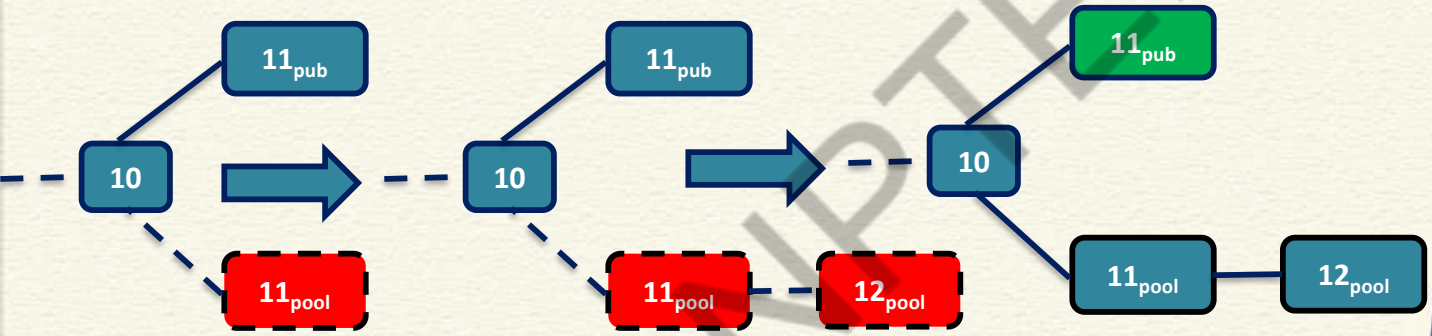
- ✓ The pool appends one block to its private branch, increasing its lead on the public branch by one
- ✓ Revenue from this block will be determined later



Selfish Mining Attack

2. Was two branches of length 1, pool finds a block

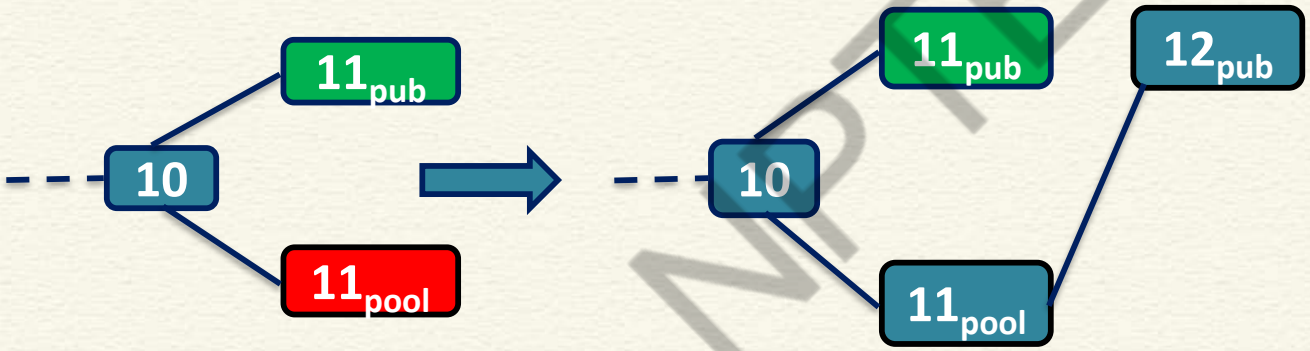
- ✓ The pool publishes its secret branch of length two
- ✓ Pool obtains a revenue of two



Selfish Mining Attack

3. Was two branches of length 1, others find a block after pool head

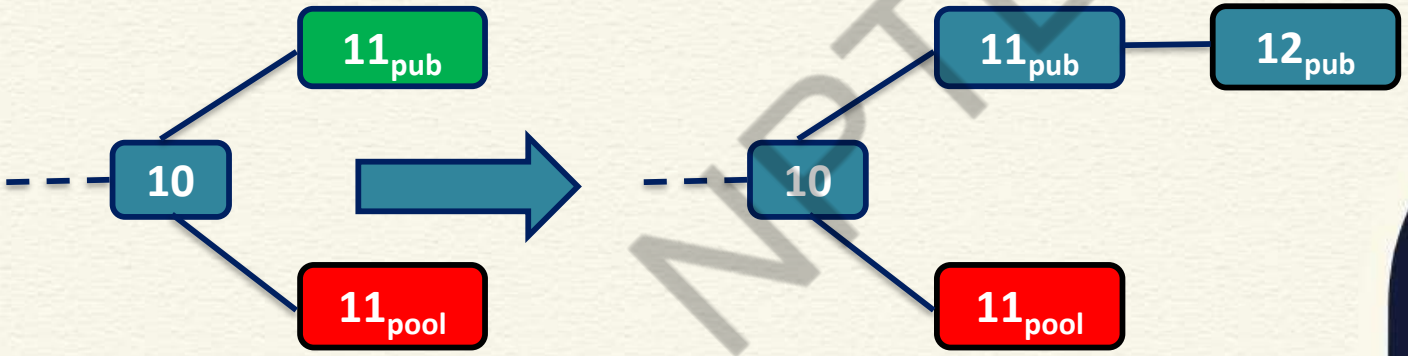
- ✓ The pool and the others obtain a revenue of one each - the others for the new head, the pool for its predecessor



Selfish Mining Attack

4. Was two branches of length 1, others find a block after others' head

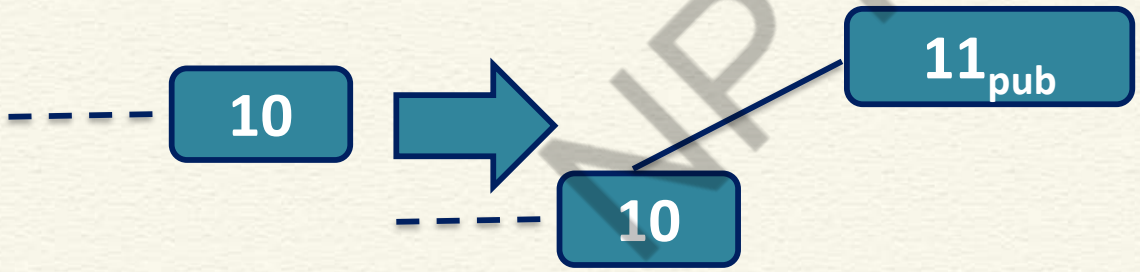
- ✓ The others obtain a revenue of two



Selfish Mining Attack

5. No private branch, others find a block

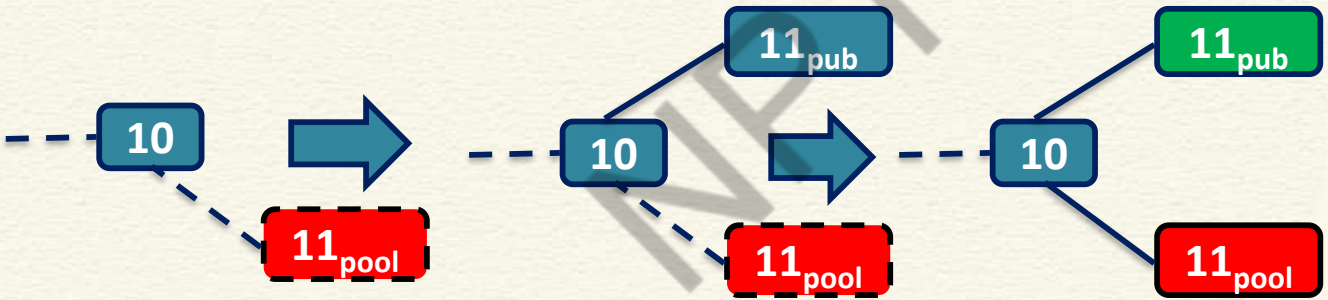
- ✓ Both the pool and the others start mining on the new head
- ✓ The others obtain a revenue of one



Selfish Mining Attack

6. Lead was 1, others find a block

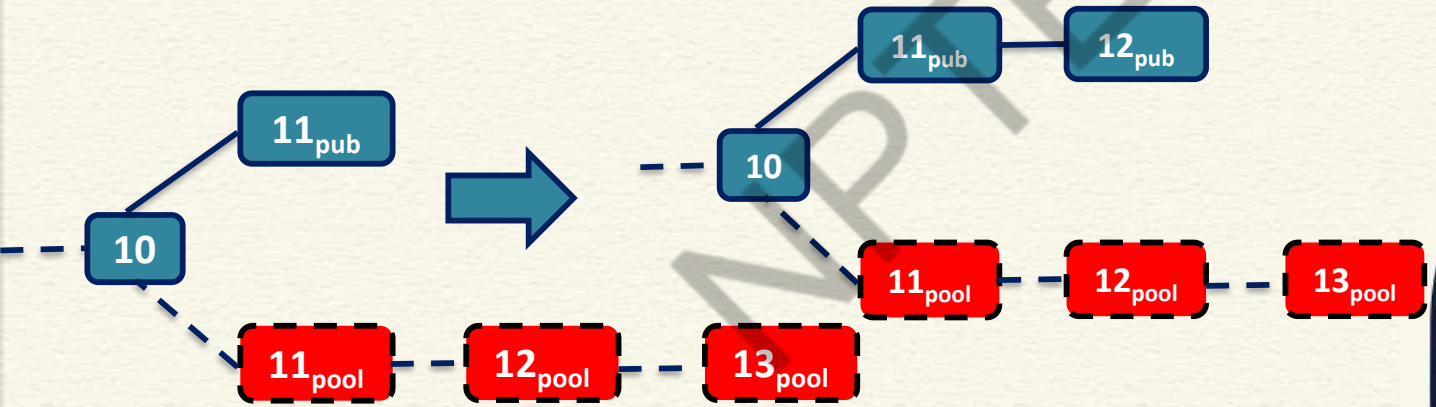
- ✓ There are two branches of length one, and the pool publishes its single secret block
- ✓ The revenue from this block cannot be determined yet



Selfish Mining Attack

7. Lead was 2, others find a block

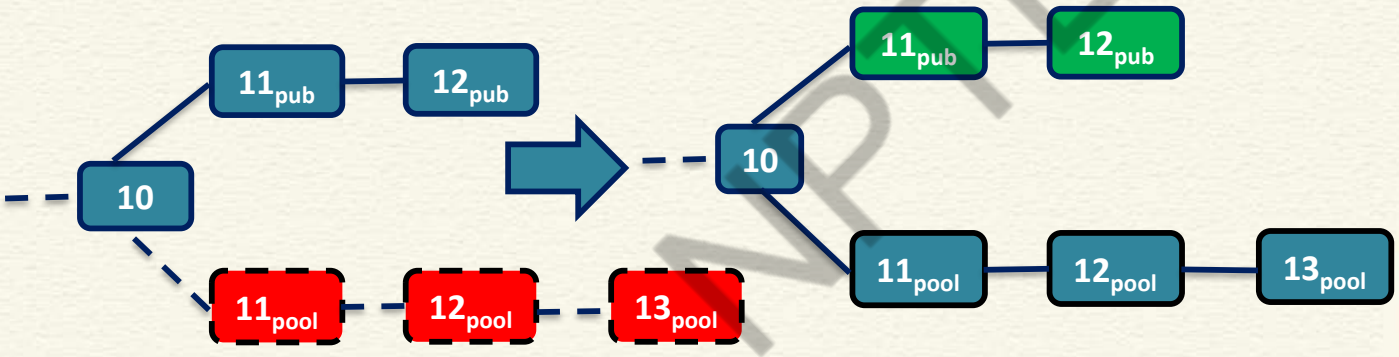
- ✓ The pool publishes its secret blocks, causing everybody to start mining at the head of the previously private branch
- ✓ Pool obtains a revenue of two



Selfish Mining Attack

7. Lead was 2, others find a block (Contd.)

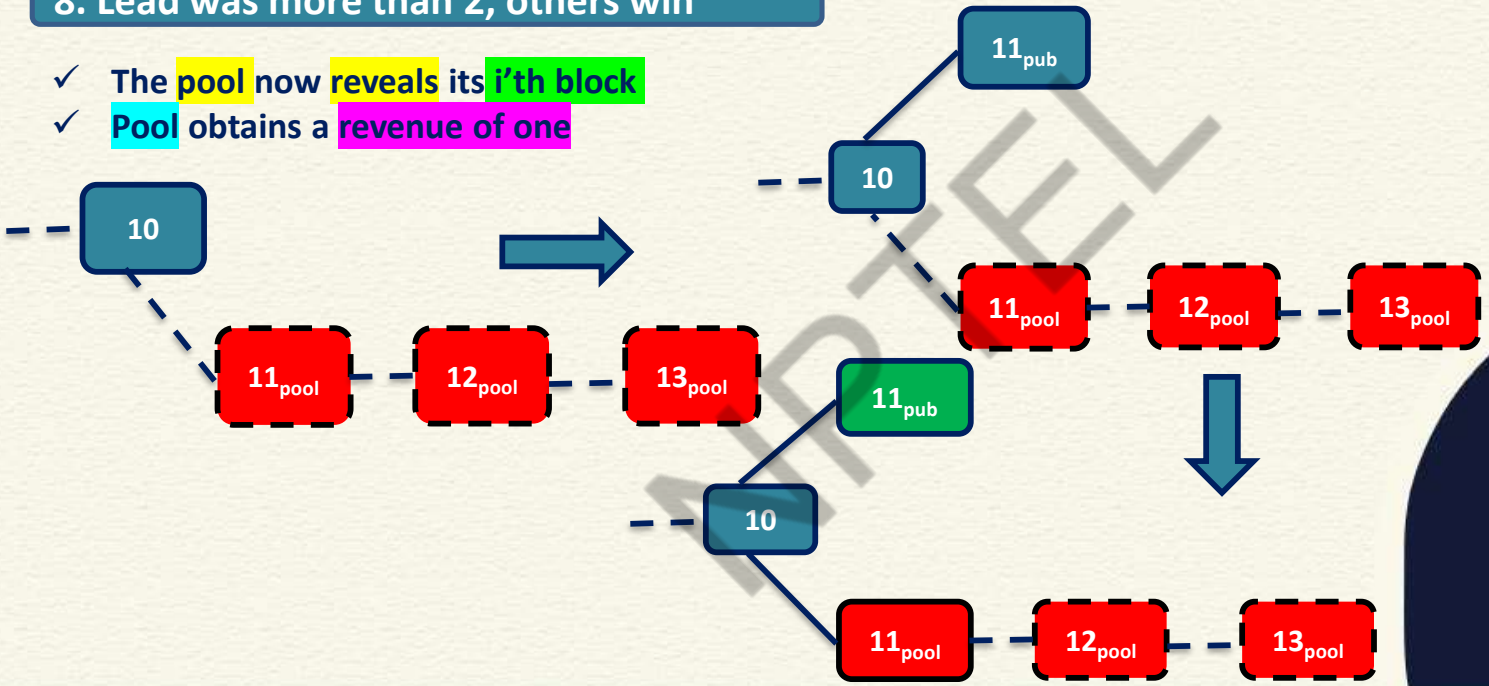
- ✓ The pool publishes its secret blocks, causing everybody to start mining at the head of the previously private branch
- ✓ Pool obtains a revenue of two



Selfish Mining Attack

8. Lead was more than 2, others win

- ✓ The pool now reveals its i'th block
- ✓ Pool obtains a revenue of one



CONCLUSIONS

- Discussed selfish mining attack in detail
- Decisions of the attacker under different conditions

REFERENCES

- **Web resources as mentioned from time to time**

NPTEL ONLINE CERTIFICATION COURSES

Blockchain and its applications

Prof. Shamik Sural
Department of Computer Science & Engineering
Indian Institute of Technology Kharagpur

Lecture 54: Blockchain Security - III

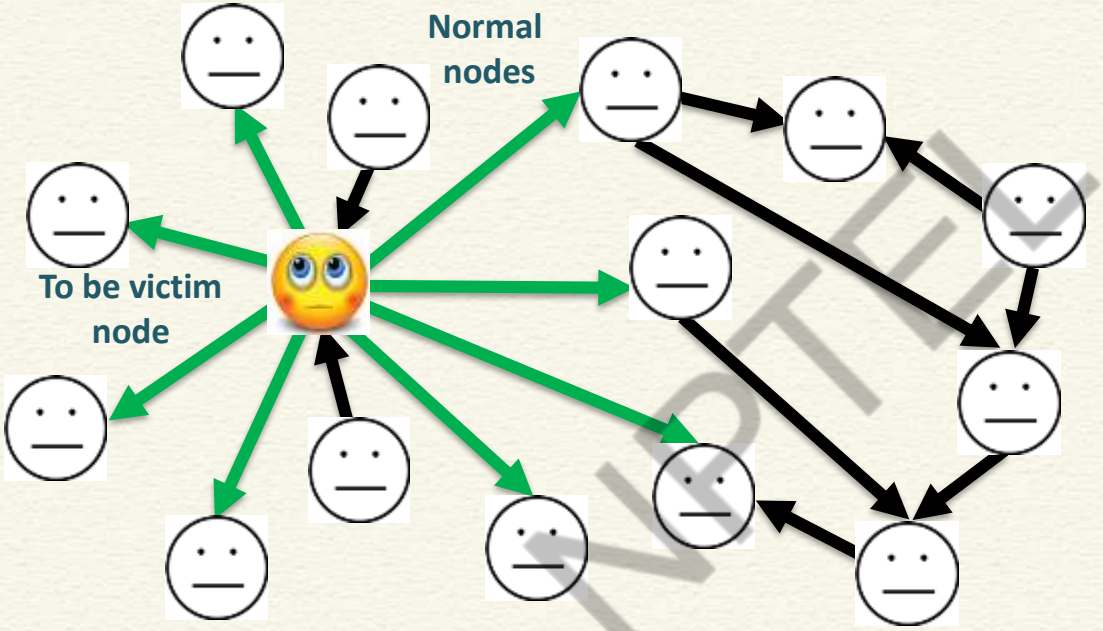
CONCEPTS COVERED

- Eclipse Attack
- Front-running Attack

KEYWORDS

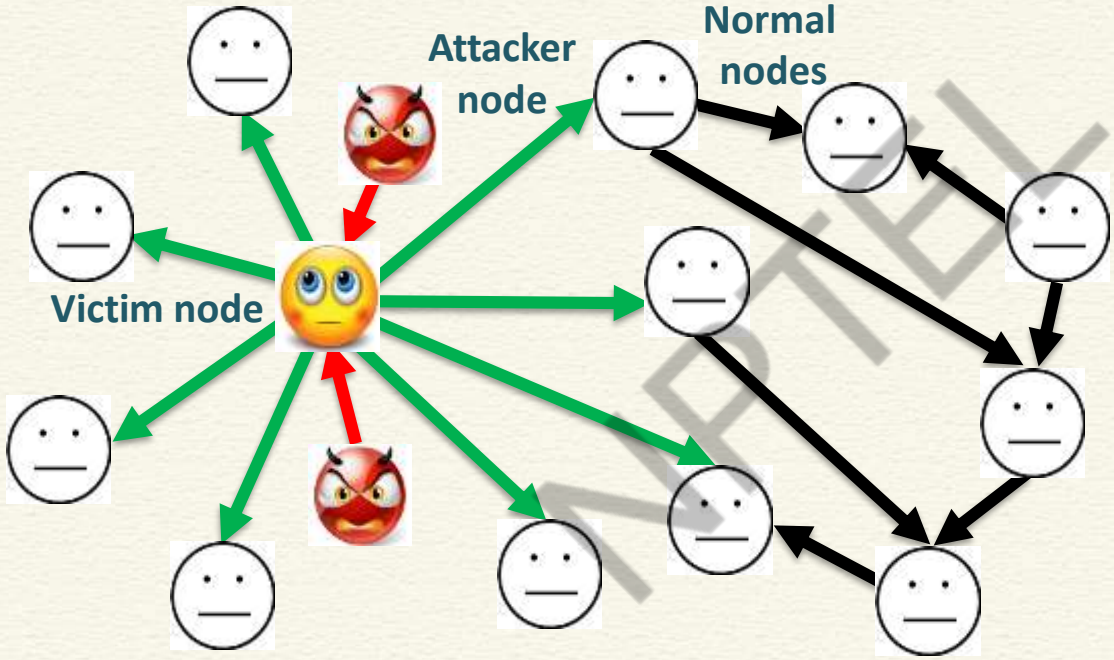
- Eclipse Attack
- Peer-to-Peer Network
- Front-running Attack
- Displacement, Insertion, Suppression

Eclipse Attack

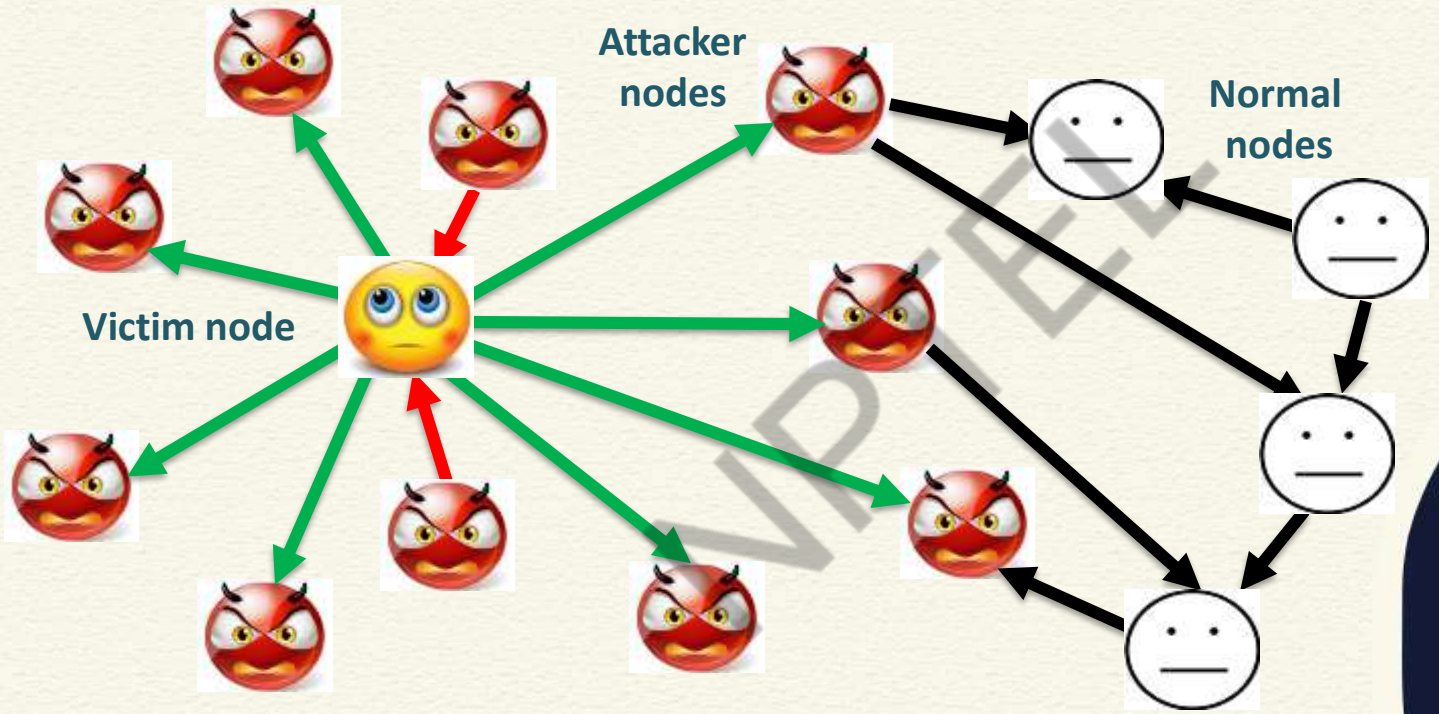


"Eclipse Attacks on Bitcoin's Peer-to-Peer Network", Ethan Heilman, Alison Kendler, Aviv Zohar and Sharon Goldberg, 24th USENIX Security Symposium, 2015

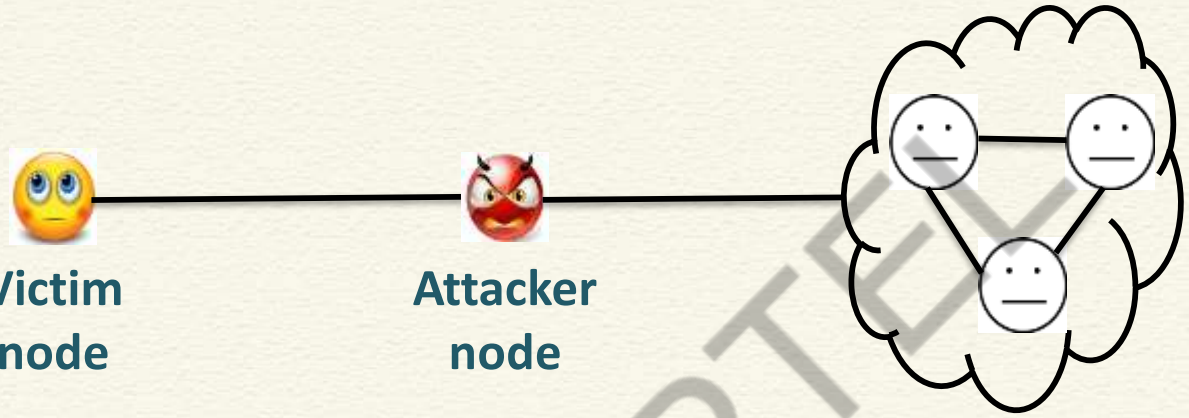
Eclipse Attack



Eclipse Attack



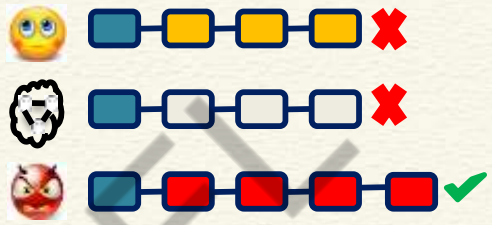
Eclipse Attack



Off-path attack - **attacker controls end-hosts**, but not key network infrastructure between the victim and the rest of the bitcoin network

Eclipse Attack

Benefit: 51% attack with 40% mining power

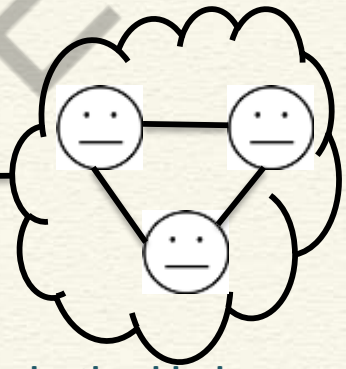


30% Hashing power

30% < 40%

40% Hashing power

40% >30%



30% Hashing power

Attacker partitions miners so they can't build on each other blocks

The attacker then outcompetes each partitioned miner

Eclipse Attack

Attacker populates the victim node's peer tables with attacker's IP addresses

Victim node restarts and loses current outgoing connections

The victim establishes all new outgoing connections to attacker IP addresses

Eclipse Attack

1. Populating of IP addresses

- ✓ Each node picks its peers from IP addresses stored in two tables
  - ➢ New table: IPs the node has heard about
  - ➢ Tried table: IPs the node peered with some point
- ✓ The tables also store a timestamp for each IP
- ✓ Each table stores the IPs in buckets
- ✓ To find an IP to make an outgoing connection to:
  1. Choose new or tired table to select from
  2. Select an IP with newest timestamp
  3. Attempt an outgoing connection to that IP

Attacker populates tables with attacker IPs so that the victim node only connects to the attacker IPs

Selection Bias: Attacker ensures its IPs are the newer one

Eclipse Attack

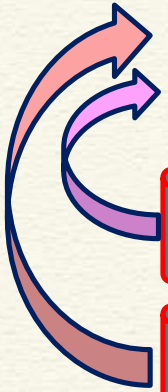
2. Restarting node event is natural?

- ✓ Software/security updates
- ✓ Packets of death/DoS attacks
- ✓ Power/network failures
- ✓ ISP outages

Eclipse Attack

3. Bucket eviction

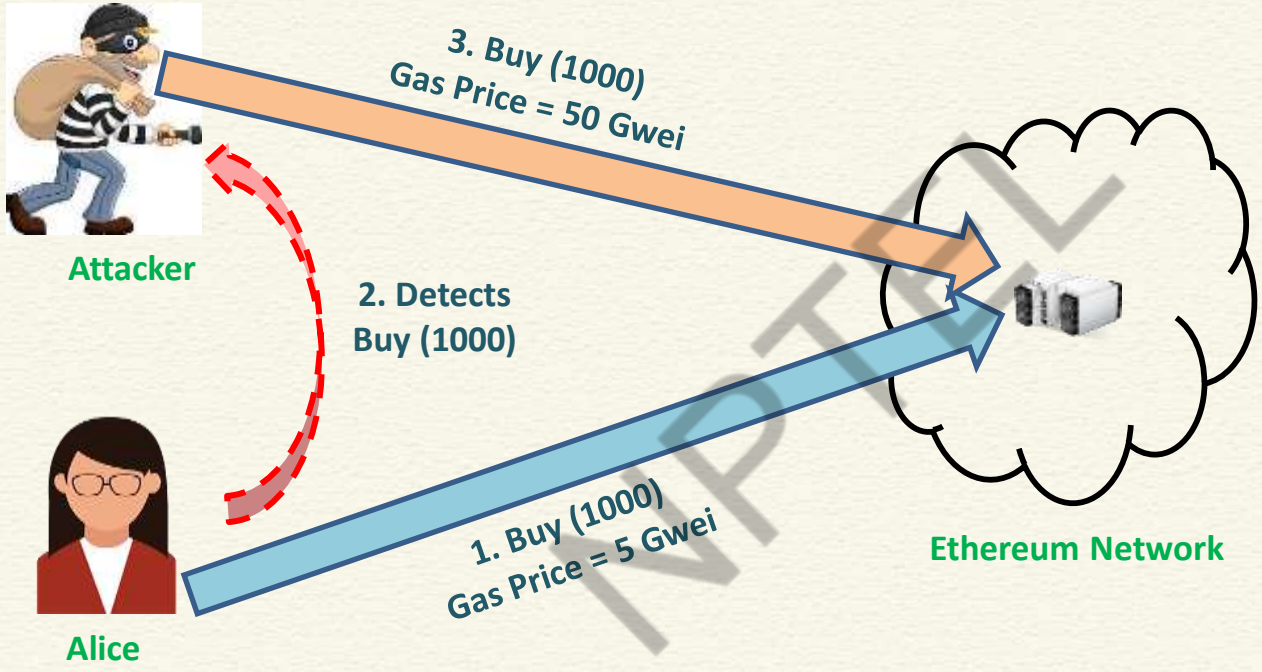
- ✓ The bucket is full, and an IP is inserted into it
  1. Randomly selects 4 IPs
  2. Delete oldest IP
  3. Insert new IP



Eviction Bias: Attacker IPs will always have the most recent timestamps

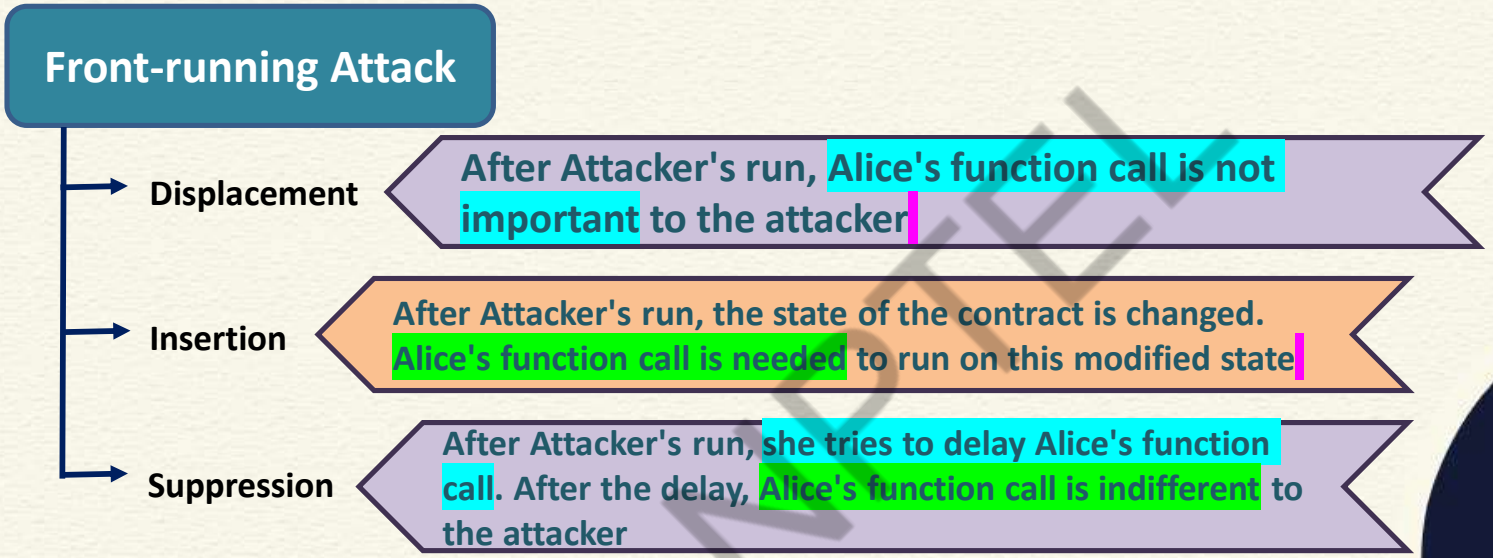
Try-Try-Again: If an attacker IP replaces another attacker IP, the evicted IP is resend and eventually replaced by honest IP

Front-running Attack



"SoK: Transparent Dishonesty: Front-Running Attacks on Blockchain", Shayan Eskandari, Seyedehmahsa Moosavi and Jeremy Clark, FC 2019 Workshops, 2020

Front-running Attack



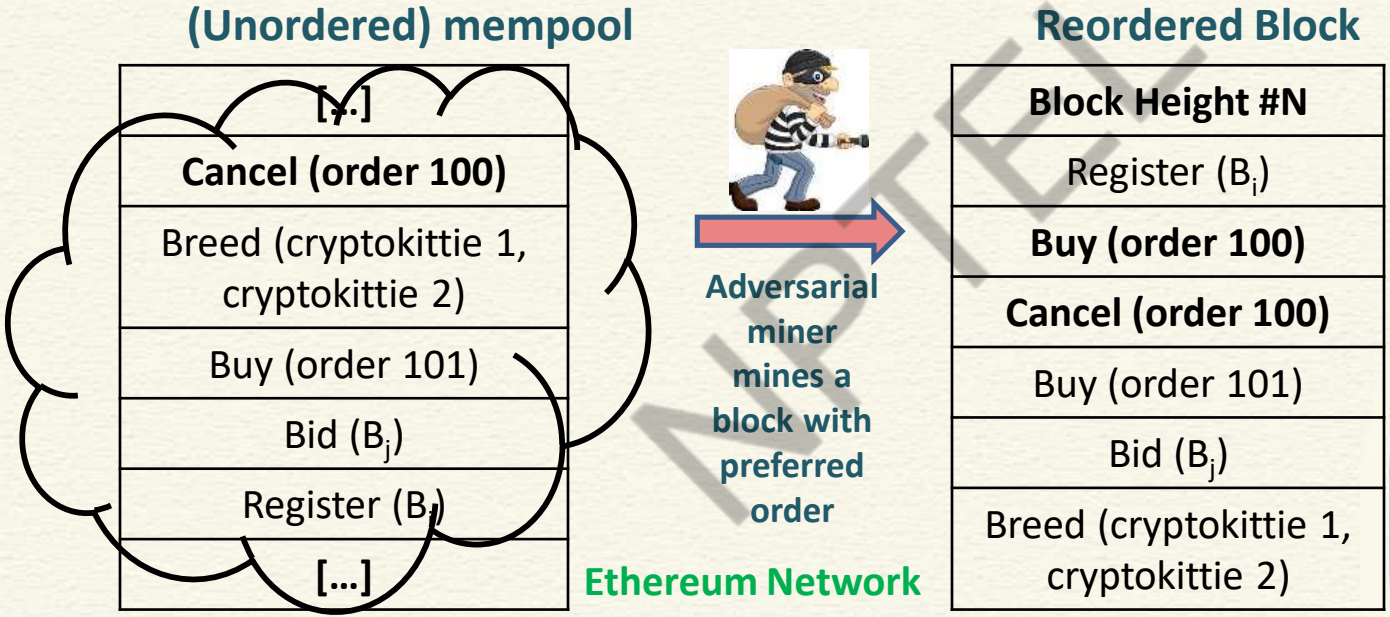
Front-running Attack

Front-running Attack
(Displacement / Insertion / Suppression)

- **Asymmetric** — Alice is trying to cancel an offer, and Attacker is trying to fulfill it first
- **Bulk** — Attacker is trying to run a large set of functions and Alice is trying to buy a limited set of shares offered by a firm on a blockchain

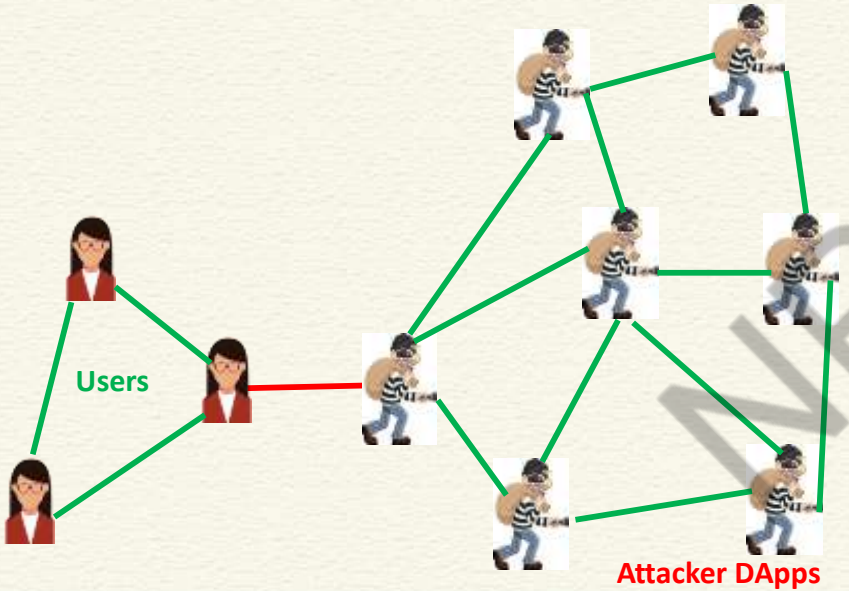
Front-running Attack

Markets and Exchanges: Spotting a profitable cancellation transaction



Front-running Attack

Gambling: Bribing miners for prioritizing themselves



- ✓ When the timer of Fomo3D game reached about 3 minutes, the winner bought 1 ticket and then sent multiple high gasPrice transactions to her own DApps
- ✓ Transactions congested the network
- ✓ Bribed miners to prioritize them ahead of any new ticket purchases in Fomo3D

CONCLUSIONS

- Described eclipse attack and front-running attack
- Importance of identifying attacks on blockchain and suggesting remedies
- Combining multiple attacks

REFERENCES

- **Web resources as mentioned from time to time**

NPTEL ONLINE CERTIFICATION COURSES

Blockchain and its applications

Prof. Sandip Chakraborty

Department of Computer Science & Engineering
Indian Institute of Technology Kharagpur

Lecture 55: Use Cases

CONCEPTS COVERED

- Blockchain use cases
- What makes a good blockchain use case?

KEYWORDS

- Use cases for enterprises
- Requirements for defining a blockchain
  - Network
  - People
  - Assets
  - Transactions

Simple Use Cases by Industry

				
Financial Services	**Public Sector**	**Retail**	**Insurance**	**Supply Chain & Logistics**
• Trade Finance • Cross currency payments • Mortgages • KYC • Cross border tax	• Asset Registration • Citizen Identity • Medical records • Medicine supply chain	• Supply chain • Loyalty programs • Information sharing (supplier – retailer)	• Claims processing • Risk provenance • Asset usage history • Claims file	• Supply chain finance • Maintenance tracking • Provenance • Supply chain compliance

What makes a good blockchain use case?

- Identifying a good blockchain use-case is not always easy!
  - However, there should always be:

1. A **business problem** to be solved
   - That cannot be more efficiently solved with other technologies
2. An identifiable **business network**
   - With Participants, Assets and Transactions
3. A need for **trust**
   - Consensus, Immutability, Finality or Provenance

Understanding the Business Problem

1. What is the specific business problem / challenge that the project will address?
   - Scope the business challenge up front
2. What is the current way of solving this business problem?
   - Understand current systems and areas for improvement
3. Assuming the business problem is large, what specific aspects of this business problem will be addressed?

Understanding the Participants

1. Who are the business network participants (organizations) involved and what are their roles?
   - If there is no business network involved, then this is not a good use case

2. Who are the specific people within the organization and what are their job roles?
   - Understand the key users in a business network.

Understanding the Participants

- Who are the participants? How many types of participants?
- How will they access and interact with the blockchain?
- Will they be peer nodes?
- Do you need web or mobile apps?
- Are gateways (such as exchanges or data providers) needed?
- Do you need to integrate to external data sources?
- Who will operate the blockchain? Who will govern/regulate the blockchain?
- What is the value/incentive for each participant to join the network?

Identities

- Do you need to know your users?
  - Pseudo-anonymous blockchain like bitcoin does not require user identities to be verified

Identities

- In most business use-cases, some form of identity is required
  - In public blockchains, an identity oracle (linked to a trusted database) could provide such information sources
    - Sources can come from governments, financial institutions or utility providers
  - In private blockchains, a gateway or controller ensures identity is verified before credentials are issued to the user
    - Decentralized identity management is also possible – we have seen that – may be the preferred way for a blockchain application

Understanding the Assets and Transactions

1. What assets are involved and what is the key information associated with the assets?

2. What are the transactions involved, between whom, and what assets are associated with transactions?
   - Understand under what business or contractual conditions assets are under, as they transfer from one owner to another.

Defining Transactions

- What types of processes need to take place in your blockchain network?
  - Invoke actions – add, delete, change, transfer
  - Query
  - Do you need to control access to these functions based on participant types or roles?

Additional Points of Understanding

1. What are the main steps in the current workflow and how are these executed by the business network participants?

2. What is the expected benefit of applying blockchain technology to the business problem for each of the network participants?

3. What legacy systems are involved? What degree of integration with the legacy systems is needed?

Conclusion

- We need to think carefully before applying blockchain directly on a problem
  - Do we really need to use blockchain?
    - What are the pros and cons of using blockchain to solve the problem?
  - Can there be a better technology?
  - Can we define the entities?
    - The business network
    - The participants, assets, and transactions