SABRE
Protecting Bitcoin against Routing Attacks

Maria Apostolaki
ETH Zürich

Joint work with Gian Marti, Jan Müller and Laurent Vanbever
Partition Attack

An adversary *splits* the Bitcoin network in two *disjoint components*
Partition attack is general, dangerous, effective, practical
Partition attack is general, dangerous, effective, practical

Any Blockchain system is vulnerable
Partition attack is general, **dangerous**, effective, practical

Any Blockchain system is vulnerable

Double-spending, Revenue Loss, DoS
Partition attack is general, dangerous, effective, practical

Any Blockchain system is vulnerable

Double-spending, Revenue Loss, DoS

50–50 partition is feasible
Partition attack is general, dangerous, effective, practical

Any Blockchain system is vulnerable

Double-spending, Revenue Loss, DoS

50–50 partition is feasible

Any network in the world is a possible attacker
In 2017 we uncovered the practicality and effectiveness of routing attacks in Bitcoin

Hijacking Bitcoin: Routing Attacks on Cryptocurrencies

https://btc-hijack.ethz.ch

Maria Apostolaki
ETH Zürich
apmaria@ethz.ch

Aviv Zohar
The Hebrew University
aviz@cs.huji.ac.il

Laurent Vanbever
ETH Zürich
lvanbever@ethz.ch

Abstract—As the most successful cryptocurrency to date, Bitcoin constitutes a target of choice for attackers. While many attack vectors have already been uncovered, one important vector has been left out though: attacking the currency via the Internet routing infrastructure itself. Indeed, by manipulating routing advertisements (BGP hijacks) or by naturally intercepting traffic, Autonomous Systems (ASes) can intercept and manipulate a large fraction of Bitcoin traffic.

This paper presents the first taxonomy of routing attacks and their impact on Bitcoin, considering both small-scale attacks, targeting individual nodes, and large-scale attacks, targeting the network as a whole. While challenging, we show that two key properties make routing attacks practical: (i) the efficiency of routing manipulation; and (ii) the significant centralization of Bitcoin in terms of mining and routing. Specifically, we find that any network attacker can hijack few (<100) BGP prefixes to isolate ~50% of the mining power—even when considering that mining pools are heavily multi-homed. We also show that on-path network attackers can considerably slow down block propagation by interfering with few key Bitcoin messages.

We demonstrate the feasibility of each attack against the deployed Bitcoin software. We also quantify their effectiveness on the current Bitcoin topology using data collected from a Bitcoin supernode combined with BGP routing data.

The potential damage to Bitcoin is worrying. By isolating parts of the network or delaying block propagation, attackers can cause

One important attack vector has been overlooked though: attacking Bitcoin via the Internet infrastructure using routing attacks. As Bitcoin connections are routed over the Internet—in clear text and without integrity checks—any third-party on the forwarding path can eavesdrop, drop, modify, inject, or delay Bitcoin messages such as blocks or transactions. Detecting such attackers is challenging as it requires inferring the exact forwarding paths taken by the Bitcoin traffic using measurements (e.g., traceroute) or routing data (BGP announcements), both of which can be forged [41]. Even ignoring detectability, mitigating network attacks is also hard as it is essentially a human-driven process consisting of filtering, routing around or disconnecting the attacker. As an illustration, it took Youtube close to 3 hours to locate and resolve rogue BGP announcements targeting its infrastructure in 2008 [6]. More recent examples of routing attacks such as [51] (resp. [52]) took 9 (resp. 2) hours to resolve in November (resp. June) 2015.

One of the reasons why routing attacks have been overlooked in Bitcoin is that they are often considered too challenging to be practical. Indeed, perturbing a vast peer-to-peer
Bitcoin is a **distributed** network of nodes (Bitcoin clients)
Bitcoin clients establish random connections
Bitcoin clients exchange **Blocks**
Blocks contain the latest transactions
Bitcoin clients exchange Blocks
Bitcoin clients exchange **Blocks** until all clients have the same view of the transactions.
What can go wrong?
Bitcoin connections are routed over the Internet
The Internet is composed of Autonomous Systems
BGP is the default Internet routing protocol
Each Bitcoin client $n$ has an IP
AS H creates a BGP advertisement for n’s IP prefix
BGP propagates advertisements in the Internet
BGP propagates advertisements in the Internet
AS I can directly reach AS H
BGP does not check the **legitimacy** of advertisements
Attacker creates a fake BGP advertisement
Attacker attracts traffic destined to AS H using BGP hijacking
Attacker attracts connections with BGP hijacking
Attacker drops connections crossing the partition
A new block in the grey zone cannot be propagated further
A new block in the grey zone cannot be propagated further
A new block in the grey zone cannot be propagated further
SABRE:
Additional overlay network that is engineered to allow clients to exchange blocks, even if the Bitcoin network is partitioned.
SABRE:

*Additional* overlay network that is engineered to allow clients to exchange blocks, even if the Bitcoin network *is partitioned*

... without the need to deploy secure routing protocols
SABRE does not affect any of the regular Bitcoin clients
SABRE is an overlay network of special Bitcoin clients.
SABRE nodes are connected to each other
Each Bitcoin client connects to at least one SABRE node
SABRE protects the Bitcoin network from partition attacks
Block is propagated via the SABRE network
The attacker might try to fight back by attacking SABRE itself.
The attacker might try to fight back by attacking SABRE itself.

Attacker knows SABRE’s locations and code.

- BGP hijacks against SABRE nodes
- Malicious requests to take down SABRE nodes
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned.
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned.

SABRE needs to...

- secure relay-to-relay connections
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned.

SABRE needs to...

- secure relay-to-relay connections
- remain reachable by Bitcoin clients
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned.

SABRE needs to…

☐ secure relay-to-relay connections

☐ remain reachable by Bitcoin clients

☐ relay blocks under any load
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned

SABRE needs to...

- secure relay-to-relay connections
- remain reachable by Bitcoin clients
- relay blocks under any load

Network Design
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned.

SABRE needs to...

- secure relay-to-relay connections
- remain reachable by Bitcoin clients
- relay blocks under any load

Network Design

Node Design
SABRE
Protecting Bitcoin against Routing Attacks

SABRE location
inherently safe locations

SABRE design
software/hardware

Deployability
deployment opportunities
SABRE
Protecting Bitcoin against Routing Attacks

SABRE location
inherently safe locations

SABRE design
software/hardware

Deployability
deployment opportunities
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned.

SABRE needs to...

- secure relay-to-relay connections
- remain reachable by Bitcoin clients
- relay blocks
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned.

SABRE needs to...

- secure relay-to-relay connections
- remain reachable by Bitcoin clients
- relay blocks under any load
SABRE selects nodes that satisfy three properties

- each node is hosted in /24 IP prefixes
- nodes are connected via financially & distance-wise optimal paths
- relay graph is k-connected
SABRE selects nodes that satisfy three properties:

- Each node is hosted in /24 IP prefixes.
- Longer prefix hijacks are not possible.
- Nodes are connected via financially & distance-wise optimal paths.
- Relay graph is k-connected.
Relays A and relay B are hosted in ASes with customer–provider relationship.
AS A receives a BGP advertisement from AS B for the prefix of relay B
Relay A sends to relay B via a direct expensive link.
BGP is a policy-based protocol, with cost playing an important role.
AS A has a malicious or compromised neighbor AS with a least expensive link
Attacker advertises AS B’s prefix to AS A
AS A prefers the path via the attacker, because it is less expensive
The attacker can **disconnect** the relays
SABRE selects nodes that satisfy three properties

- Each node is hosted in /24 IP prefixes.
- Nodes are connected via financially & distance-wise optimal paths.
- No strictly more preferred path exists.
- Relay graph is k-connected.
Relays A, B are hosted in ASes with a more cost effective agreement
Attacker’s advertisement is less preferred, thus attacker cannot discontent the relays
Agreements can be revoked, link can be cut …
Peering agreement can be revoked, link can be cut … Relay A will inevitably send traffic via ASC
SABRE selects nodes that satisfy three properties

- each node is hosted in /24 IP prefixes
- nodes are connected via financially & distance-wise optimal paths
- relay graph is k-connected
- relay connectivity is not disrupted by any k−1 cuts
2-k connected graph retains connectivity even if one peering link is cut
If the link between relays A and B is cut
If the link between relays A and B is cut, relays A, B can still exchange blocks via the relay C.
If the link between relays A and B is cut, relays A, B can still exchange blocks via the relay C.
If the link between relays A and B is cut, relays A, B can still exchange blocks via the relay C.
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned

SABRE needs to...

- [✓] secure relay-to-relay connections
- [ ] remain reachable by Bitcoin clients
- [ ] relay blocks

Node Design
SABRE positions nodes s.t. most clients are protected from each potential attacker by at least one relay node

see paper for more details
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned.

SABRE needs to...

- secure relay-to-relay connections
- remain reachable by Bitcoin clients
- relay blocks under any load
We evaluate SABRE’s network design by its effectiveness against two attack types:

- Network-wide attacks
- Node-level attacks
We evaluate SABRE’s network design by its effectiveness against two attack types

- Network-wide attacks
- Node-level attacks
We evaluate SABRE’s network design by its effectiveness against two attack types.

- Network-wide attacks
- Node-level attacks
We evaluate SABRE’s network design by its effectiveness against two attack types:

- **Network-wide attacks**
  - What is the largest partition each single AS can create?

- **Node-level attacks**
  - How many clients are protected against isolation?
What is the largest partition each single AS can create?
What is the largest partition each single AS can create?

- current network

  any single AS in the world can create partitions of 90% of the clients
What is the largest partition each single AS can create?

- Current network
  - Any single AS in the world can create partitions of 90% of the clients

- 6 SABRE nodes
  - Only 3% of ASes in the world can create partitions of 15% of the clients

See paper for more results.
We evaluate SABRE’s network design by its effectiveness against two attack types:

- **Network-wide attacks**
  - What is the largest partition each single AS can create?

- **Node-level attacks**
  - How many clients are protected against isolation?
How many clients are protected against isolation?
How many **clients** are protected against isolation?

- **current network**
  
  at most **10%** of Bitcoin clients are protected from **50%** of ASes
How many clients are protected against isolation?

- current network
  - at most 10% of Bitcoin clients are protected from 50% of ASes

- 6 SABRE nodes 5–connected
  - 89.5% of Bitcoin clients are protected from 92.5% of ASes

see paper for more results
SABRE
Protecting Bitcoin against Routing Attacks

SABRE location
inherently safe locations

SABRE design
software/hardware

Deployability
deployment opportunities
SABRE is an additional overlay network which allows communication, even if the Bitcoin network is partitioned.

SABRE needs to...

- secure relay-to-relay connections
- remain reachable by Bitcoin clients
- relay blocks under any load
Two ways to deploy a SABRE node

Private deployment

Public deployment
Two ways to deploy a SABRE node

Private deployment
Serving few predefined clients

Public deployment
Private SABRE nodes need not scale

SABRE nodes need to

- establish connection to a predefined set of IPs
- receive and relay blocks
- be unreachable for unknown clients
Private SABRE nodes need not scale

SABRE nodes need to

- establish connection to a predefined set of IPs
- receive and relay blocks
- be unreachable for unknown clients

current Bitcoin client implementation hosted in a VM is sufficient
Two ways to deploy a SABRE node

- **Private deployment**
  - Serving few *predefined* clients

- **Public deployment**
  - Serving *all* Bitcoin clients
Public SABRE nodes need to scale

SABRE nodes need to:

- maintain thousands of connections
- receive, verify and relay blocks fast
- protect against spoofing and malicious request
Public SABRE nodes need to scale

SABRE nodes need to

- maintain thousands of connections
- receive, verify and relay blocks fast
- protect against spoofing and malicious request

Simple software implementation would not suffice
SABRE can leverage programmable data planes

SABRE DP
SABRE DP allows relay nodes to deal with high malicious or benign load
SABRE DP allows relay nodes to deal with high malicious or benign load is faster than any server optimization.

can serve few Billions of packets per second

NetChain: Scale-Free Sub-RTT Coordination
NDSI 2018
SABRE DP allows relay nodes to deal with high malicious or benign load is faster than any server optimization

protects against malicious requests

Dynamic Black/White lists anti-spoofing mechanism & DoS protection
SABRE DP allows relay nodes to deal with high malicious or benign load

is faster than any server optimization

protects against malicious requests

minimum software interaction

almost all clients served directly from hardware
Not all operations can be done in hardware
Not all operations can be done in hardware
SABRE node has both software and hardware parts

control plane
software

data plane
hardware

SABRE

UDP connection

Bitcoin (TCP) connection
SABRE
Protecting Bitcoin against Routing Attacks

SABRE location
inherently safe locations

SABRE design
software/hardware

Deployability
deployment opportunities
SABRE’s deployment is practical
SABRE’s deployment is practical

bootstrap with a software-only SABRE

decreased cost
allows private deployments
SABRE’s deployment is practical

bootstrap with a software-only SABRE

multiple SABRE relays can co-exist

each party (e.g. pool) can deploy their own SABRE without coordination
SABRE’s deployment is practical

bootstrap with a software-only SABRE

multiple SABRE relays can co-exist

community’s consensus is not required

clients can connect to both relays and regular clients
SABRE’s deployment is practical

bootstrap with a software-only SABRE

multiple SABRE relays can co-exist

community’s consensus is not required

network design applies to other relays

e.g., FIBRE, FALCON can relocate relays following SABRE location algorithm
SABRE
Protecting Bitcoin against Routing Attacks

SABRE location
inherently safe locations

SABRE design
software/hardware

Deployability
deployment opportunities
SABRE can protect Bitcoin from partitions by placing few relay nodes in selected locations.

SABRE can operate seamlessly under high load by serving clients directly in hardware.

SABRE can be partially deployed and benefit early adopters e.g., each pool can deploy SABRE in software.