PhD Thesis

EFFICIENCY AND SECURITY IN PEER-TO-PEER STREAMING PROTOCOLS

DOCTORADO EN INFORMÁTICA (RD99/11)
DEPARTMENT OF INFORMATICS
UNIVERSITY OF ALMERÍA
SPAIN

Author

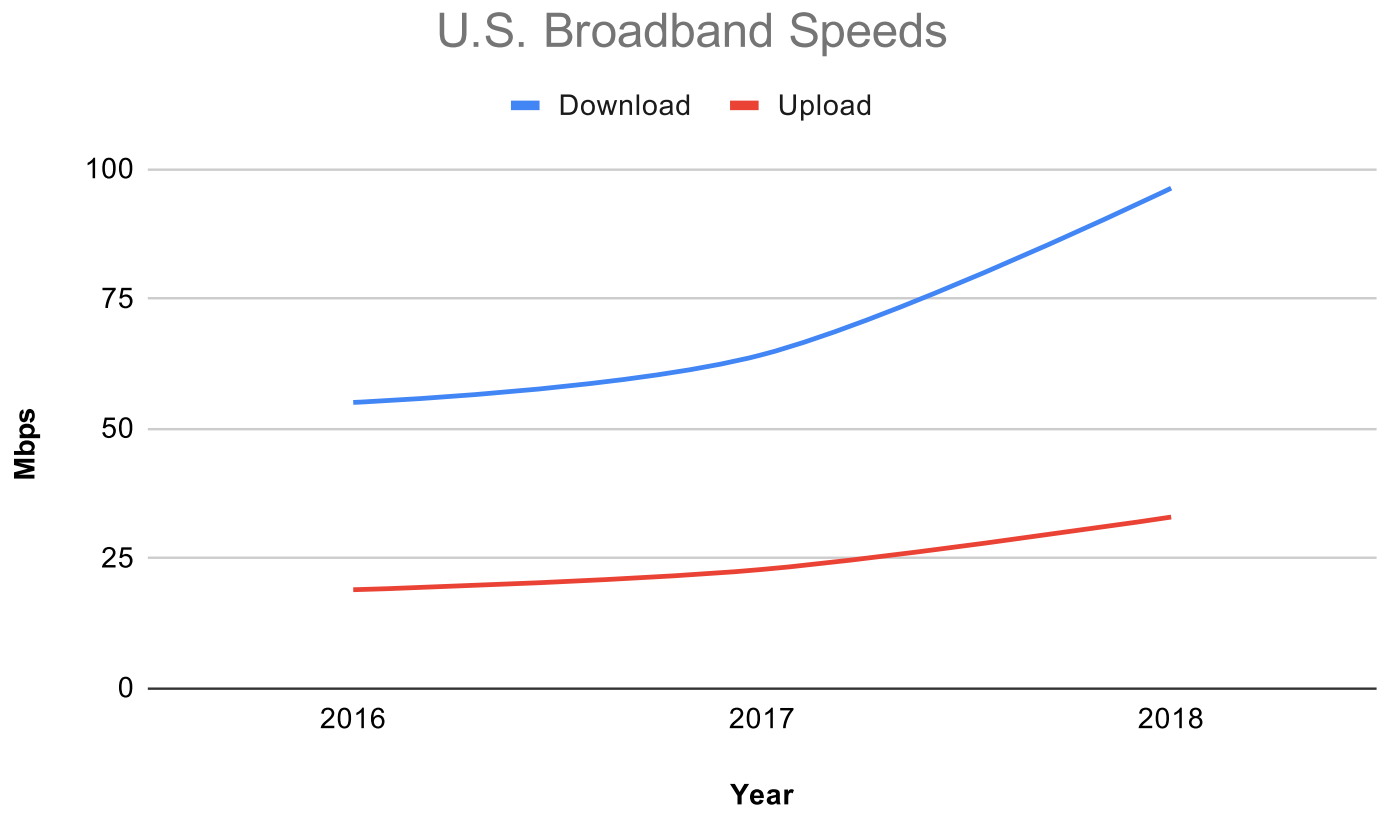
CRISTÓBAL MEDINA LÓPEZ

Supervisors

DR. LEOCADIO GONZÁLEZ CASADO
DR. VICENTE GONZÁLEZ RUIZ

November 11, 2019

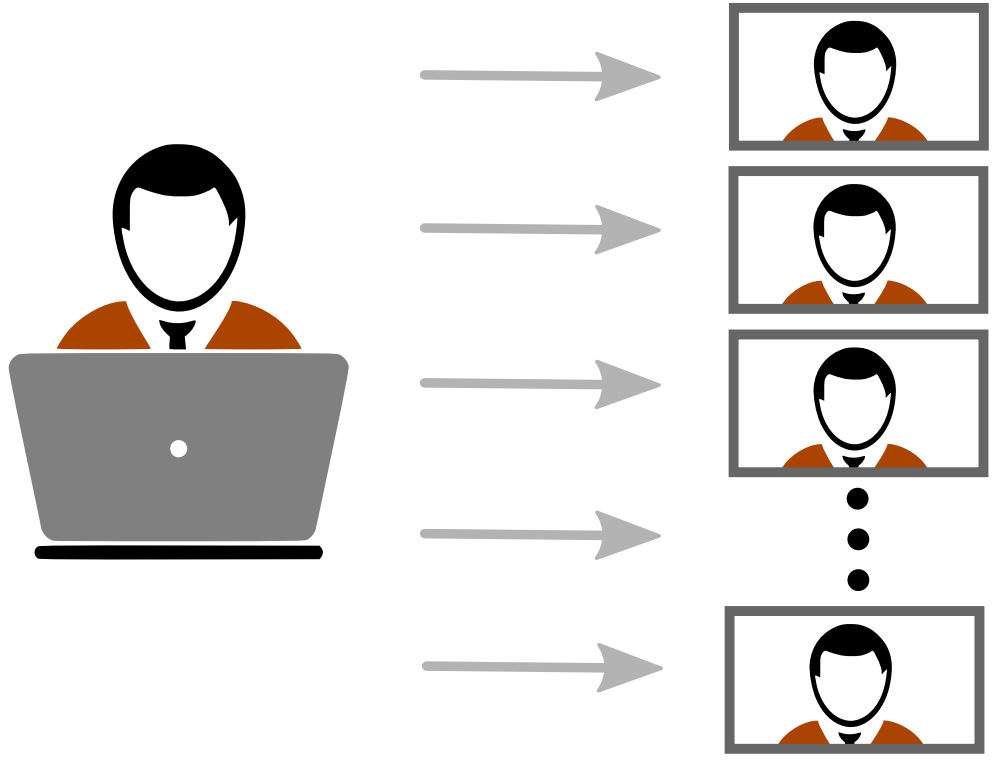
INTERNET HAS CHANGED THE WAY WE COMMUNICATE



Source: https://www.speedtest.net/reports/

HIGH QUALITY STREAMING TO THE LARGEST NUMBER OF USERS

with lowest cost in the shortest possible time



OUTLINE

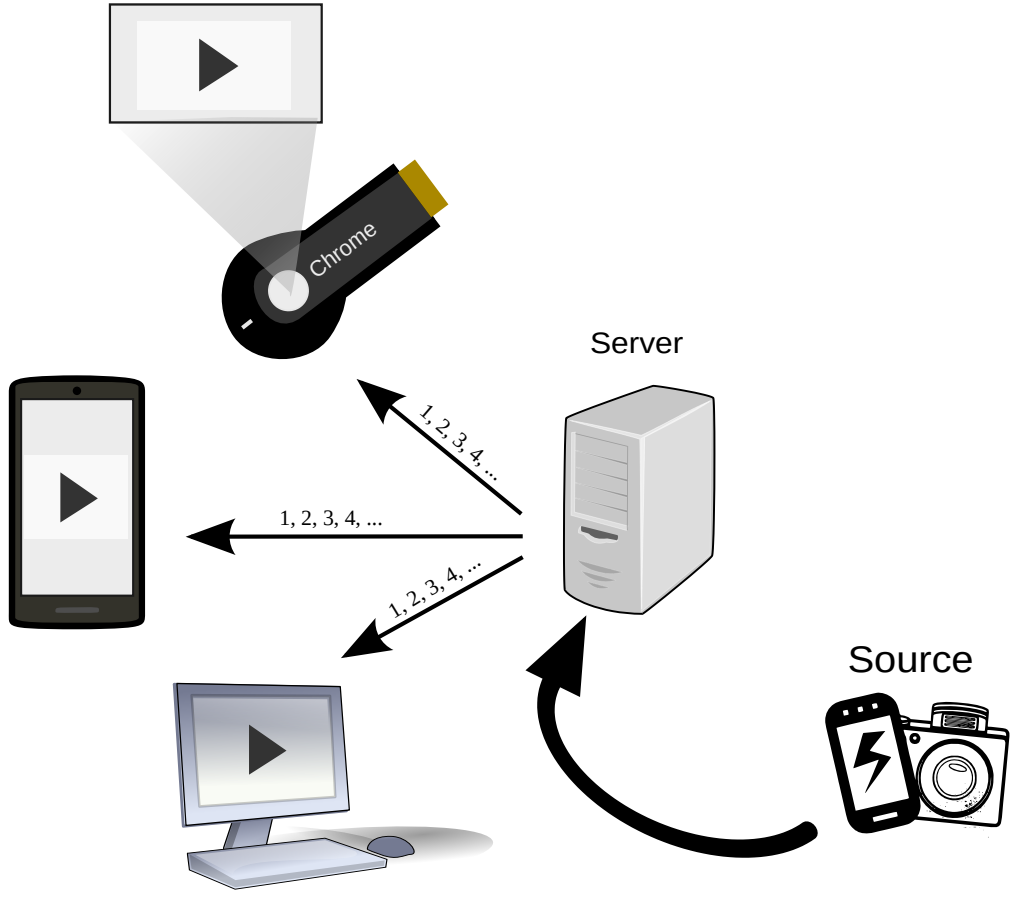
- **INTRODUCTION**
  - Background
  - Research Questions
- **PEER TO PEER STRAIGHFORWARD PROTOCOL**
  - An application-layer protocol that provides real-time broadcasting on the Internet
- **SECURITY**
  - Pollution Attacks
  - Strategies applied to Traditional Networks
  - Strategies applied to Software Defined Networks
- **EFFICIENCY**
  - NAT Traversal using Collaborative Port Prediction
  - Runing P2PSP on Embeded Devices
- **CONCLUSIONS**
  - Key Results and Future Work

OUTLINE

- **INTRODUCTION**
  - Background
  - Research Questions
- **PEER TO PEER STRAIGHFORWARD PROTOCOL**
  - An application-layer protocol that provides real-time broadcasting on the Internet
- **SECURITY**
  - Pollution Attacks
  - Strategies applied to Traditional Networks
  - Strategies applied to Software Defined Networks
- **EFFICIENCY**
  - NAT Traversal using Collaborative Port Prediction
  - Runing P2PSP on Embeded Devices
- **CONCLUSIONS**
  - Key Results and Future Work

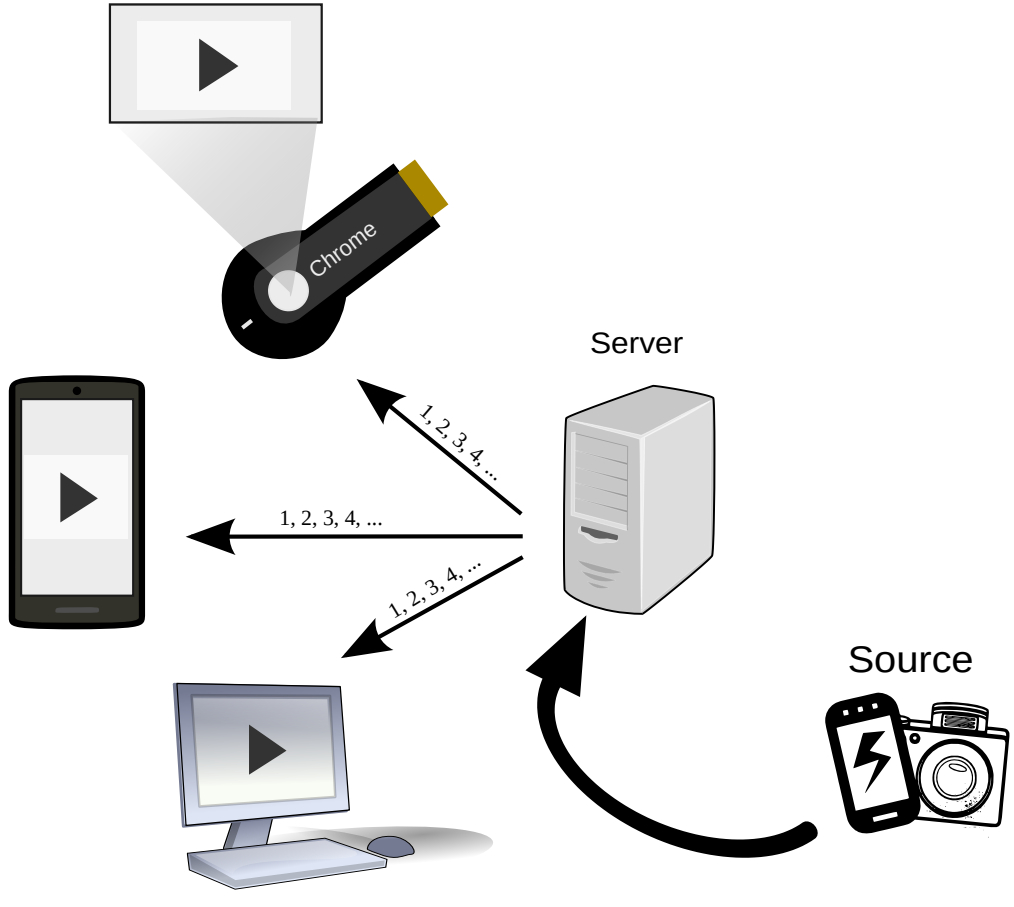
MOST POPULAR SOLUTIONS

CLIENT-SERVER MODEL



MOST POPULAR SOLUTIONS

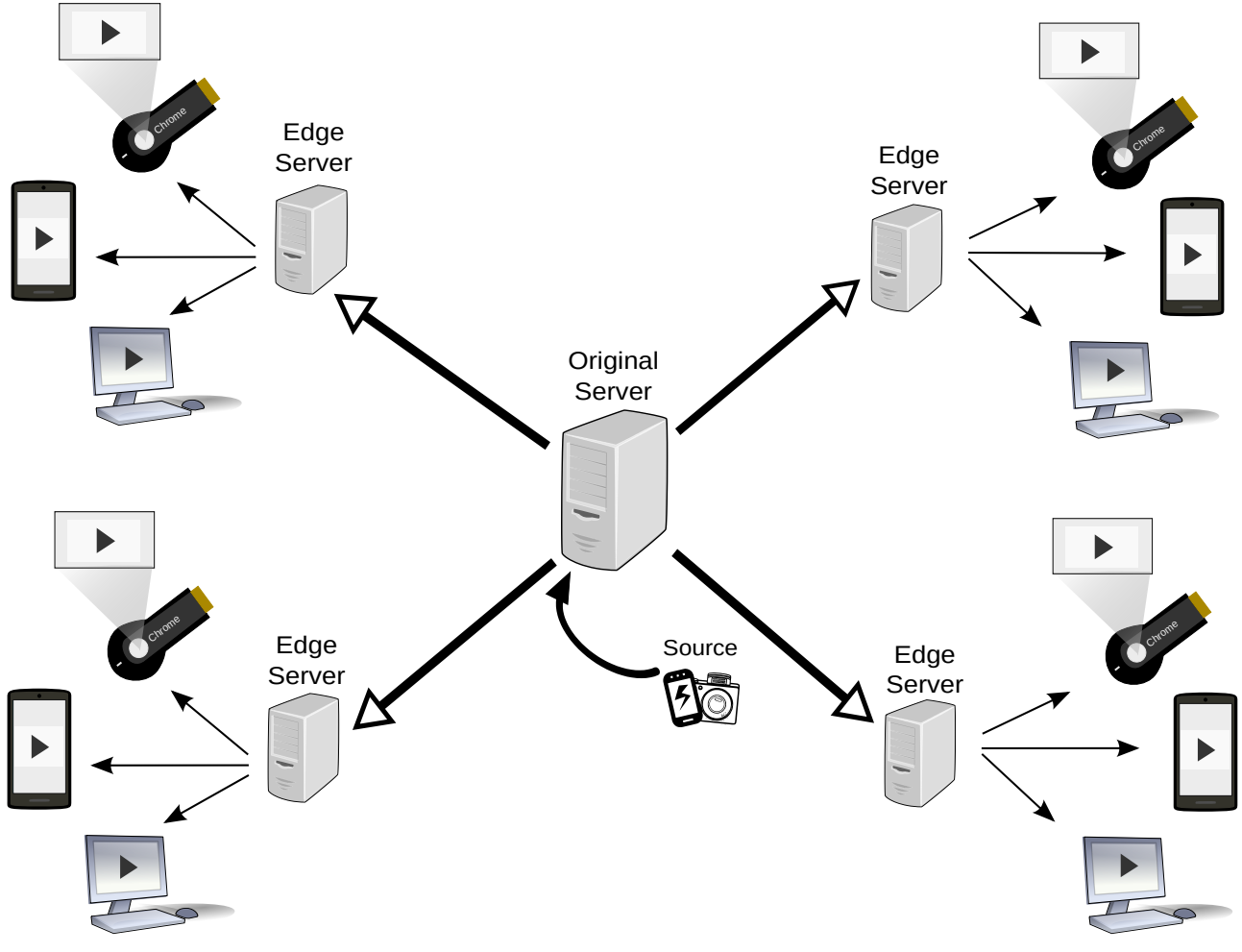
CLIENT-SERVER MODEL



It is not scalable without replication

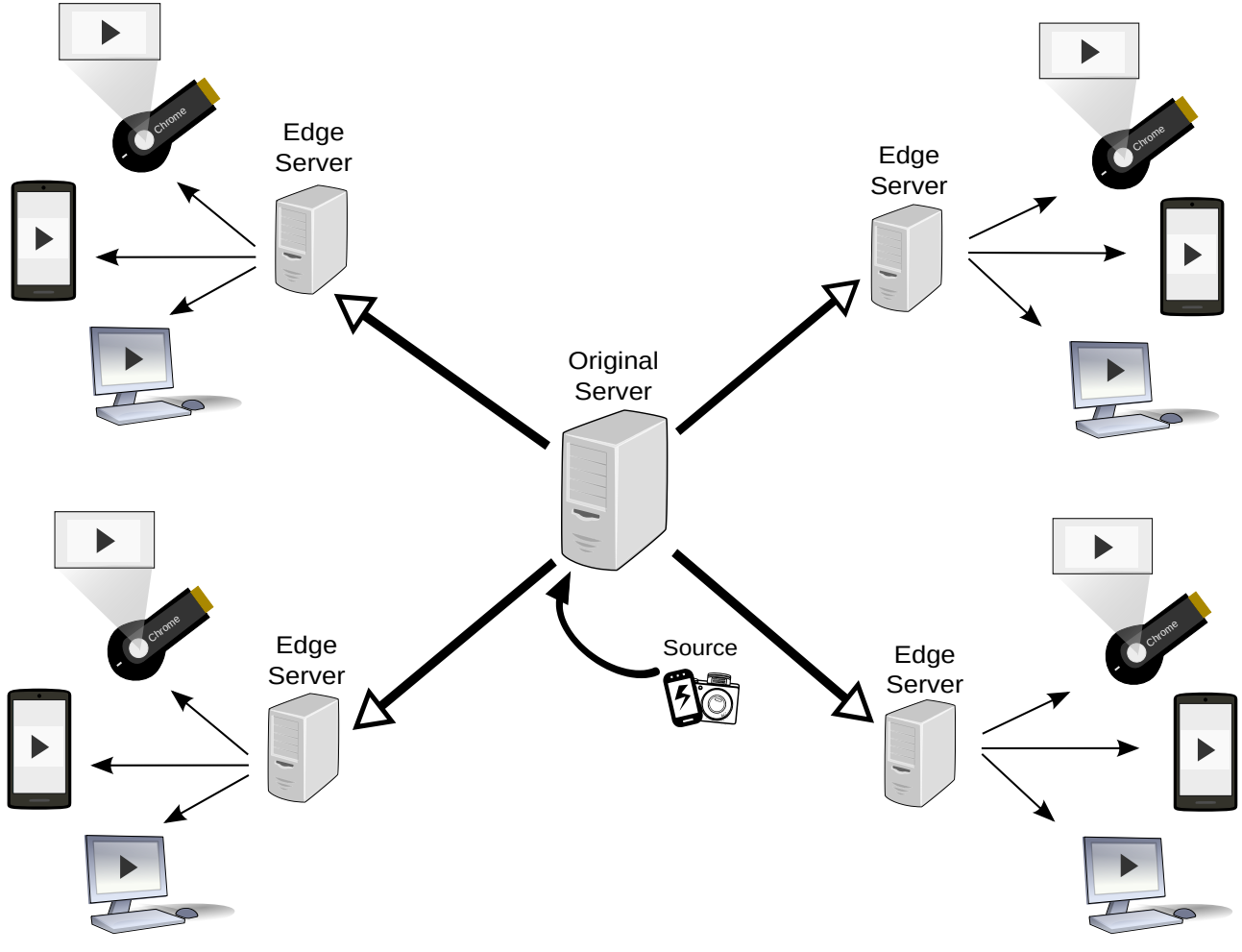
MOST POPULAR SOLUTIONS

CONTENT DELIVERY NETWORKS



MOST POPULAR SOLUTIONS

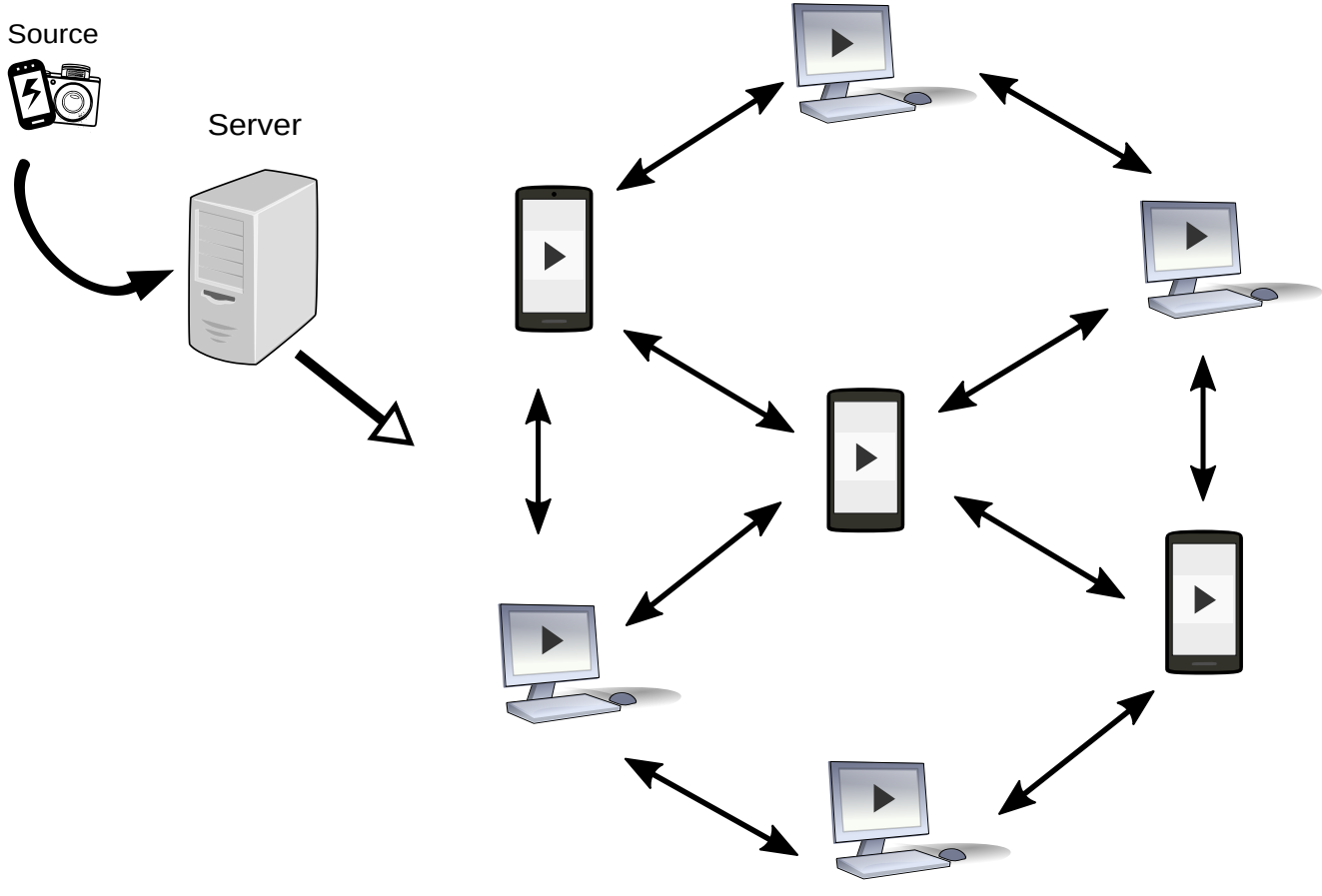
CONTENT DELIVERY NETWORKS



Significant increase in the cost of the infrastructure

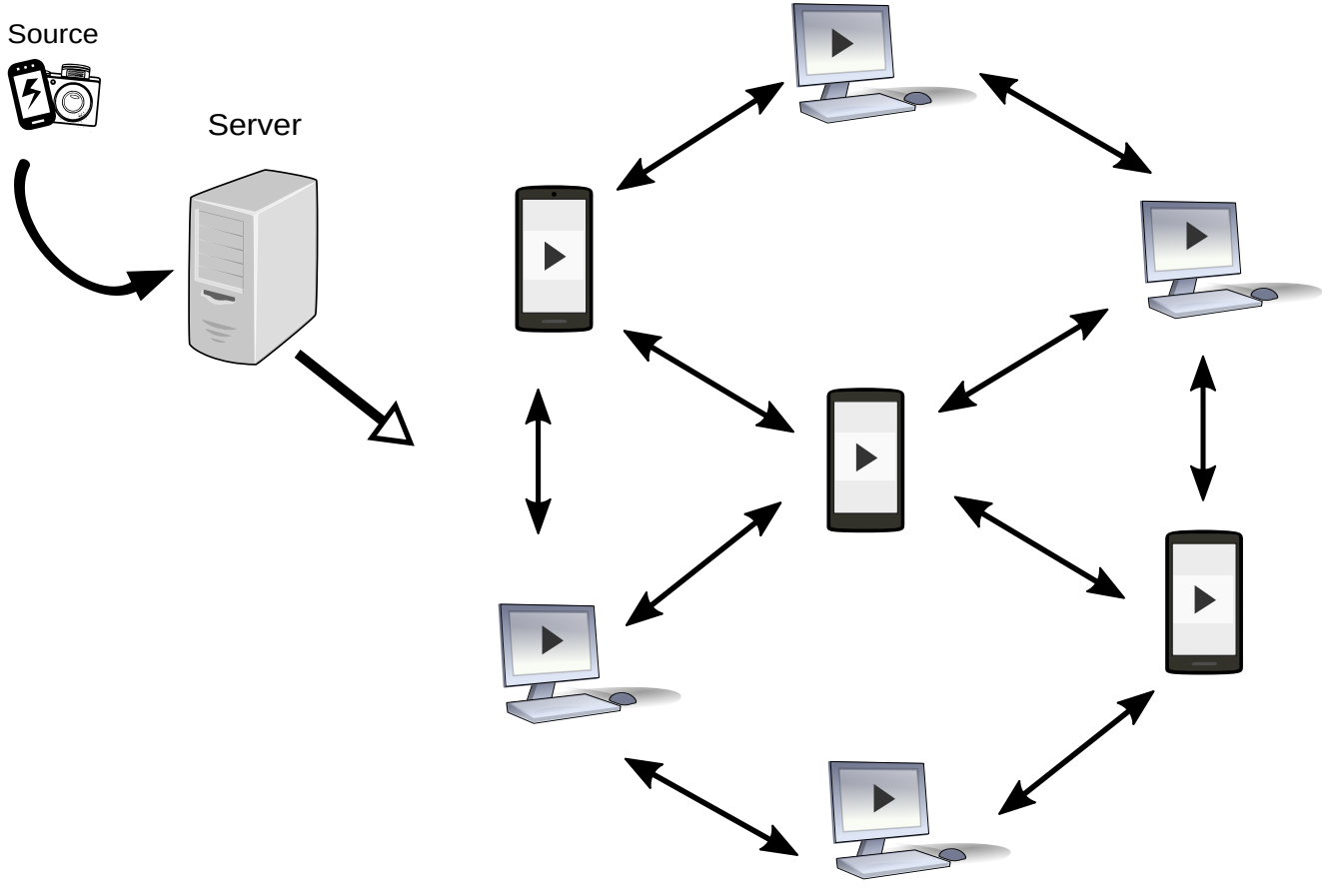
ALTERNATIVE SOLUTION

PEER-TO-PEER NETWORKS



ALTERNATIVE SOLUTION

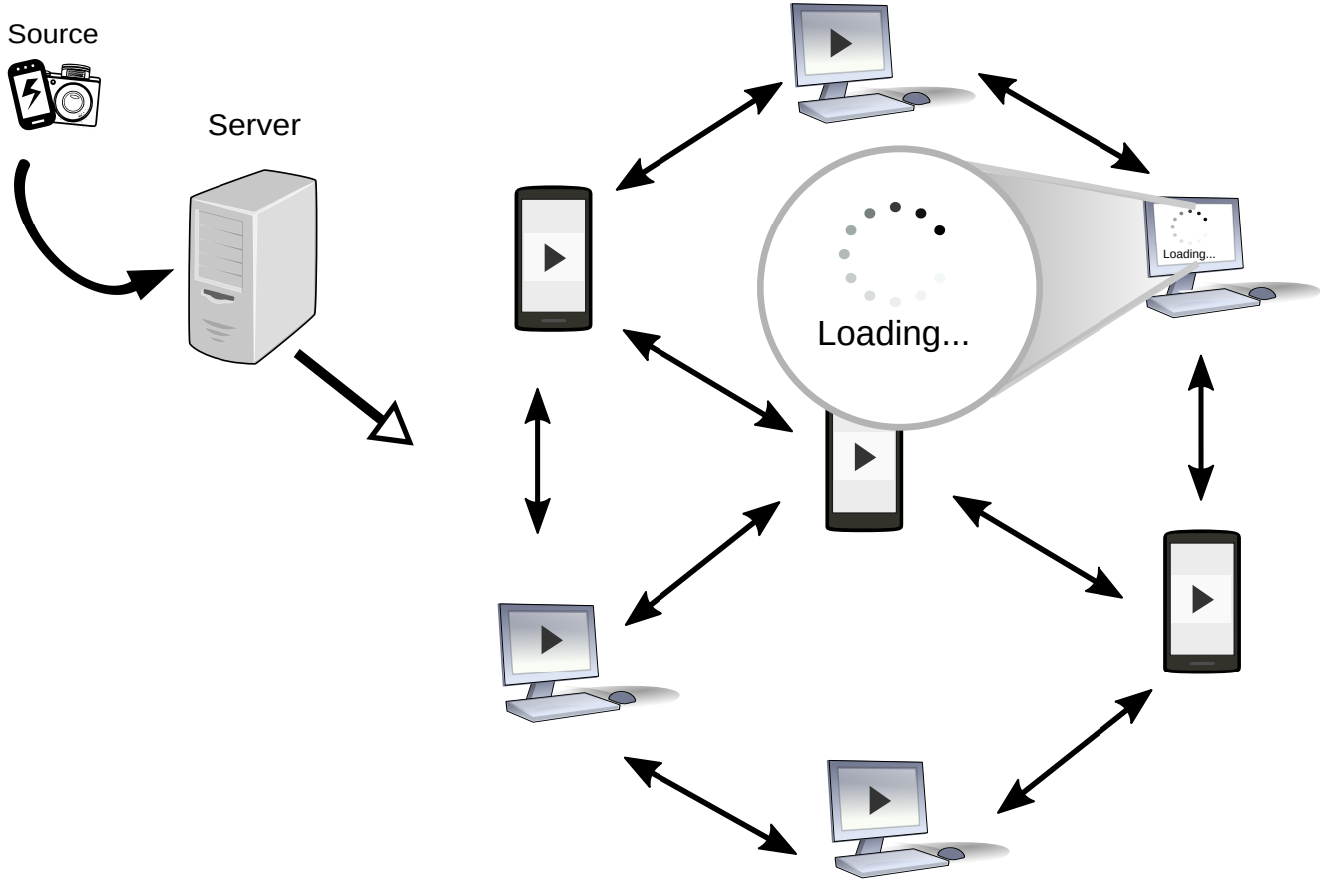
PEER-TO-PEER NETWORKS



Latency, Security, Network Limitations, Client Resources

PEER-TO-PEER NETWORKS

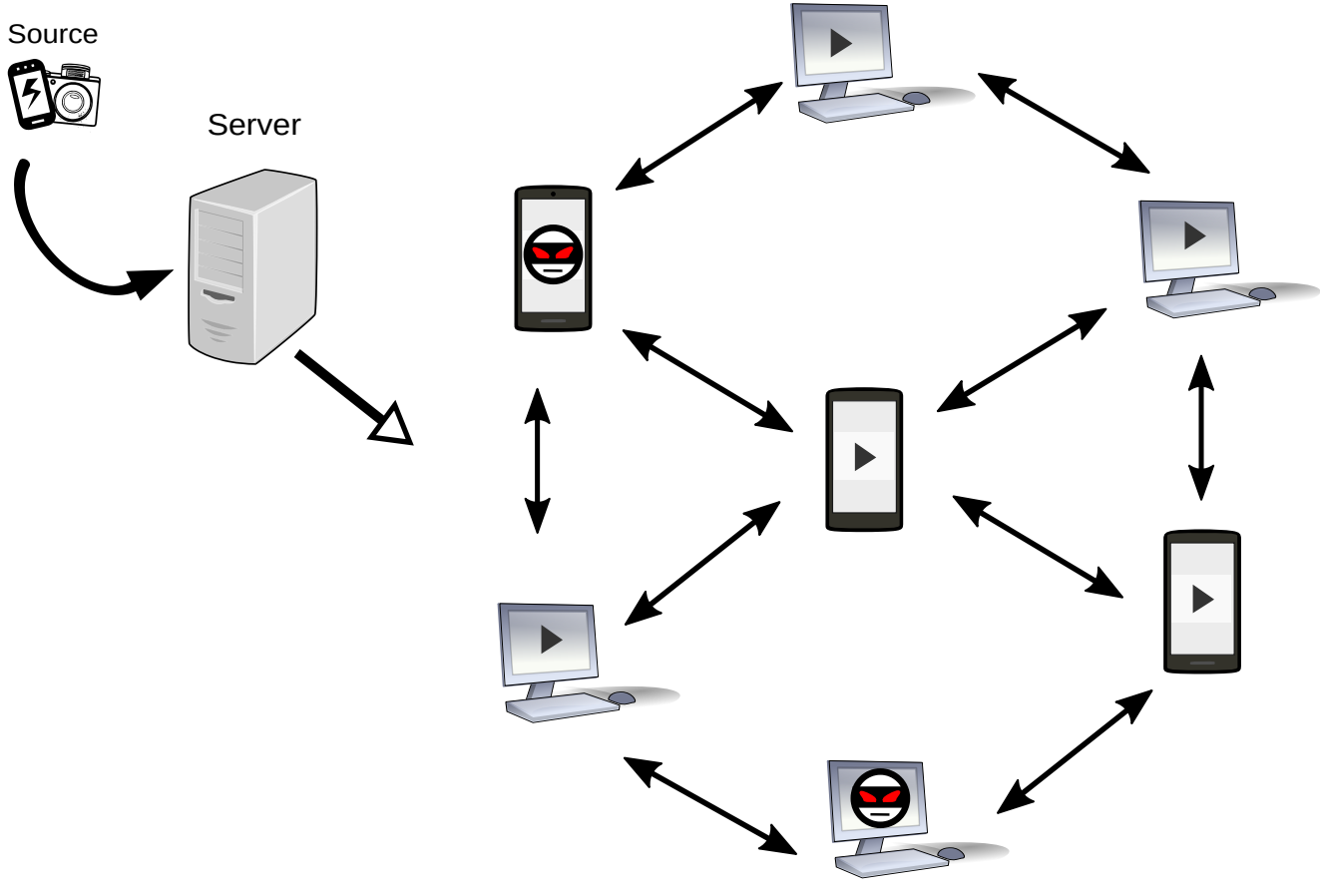
PROBLEMS: LATENCY



Because of its distributed approach, sometimes there is a considerable delay from when the content is generated until users play it.

PEER-TO-PEER NETWORKS

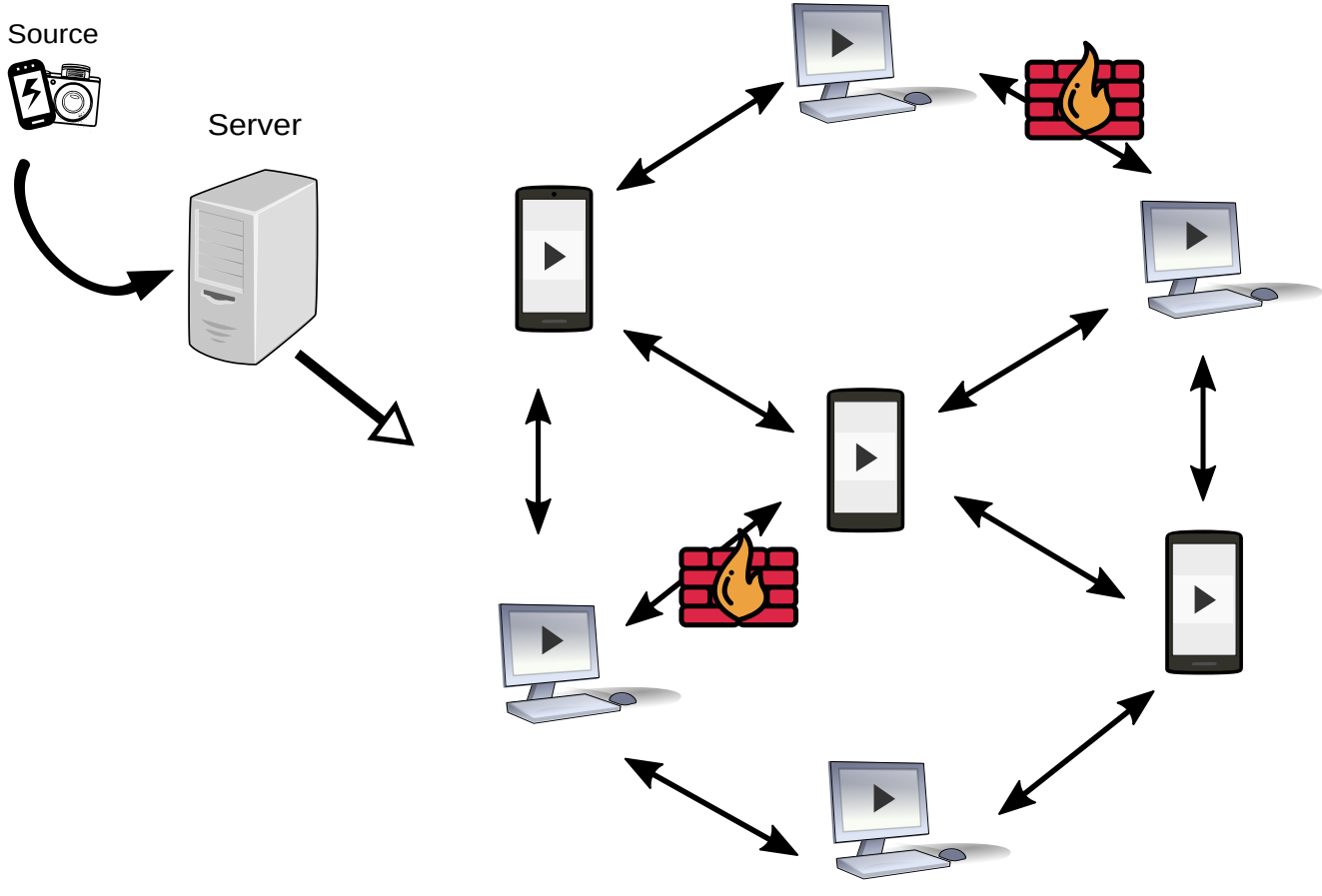
PROBLEMS: SECURITY



The anonymity provided by most P2P networks leads to malicious peers attacking in different ways: denial of service attacks, poisoning of content, avoiding sharing their resources, etc.

PEER-TO-PEER NETWORKS

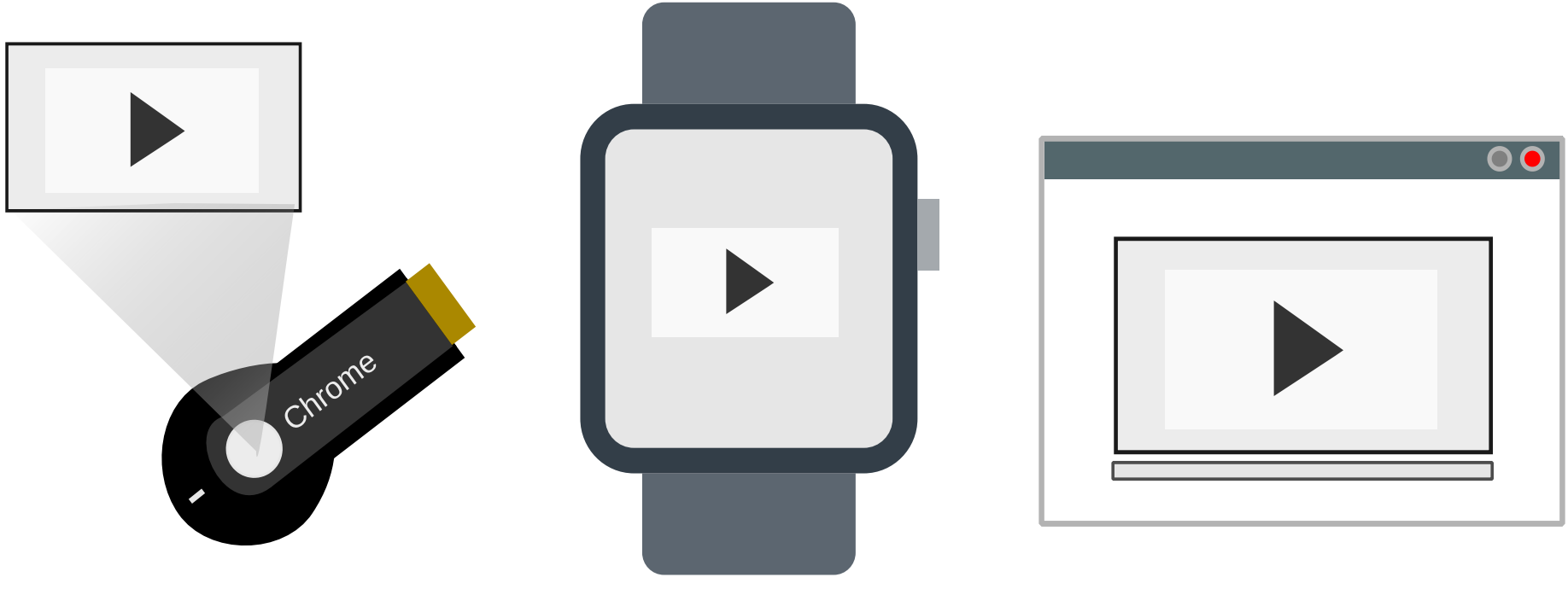
PROBLEMS: NETWORK LIMITATIONS



Limitations in networks caused by peers behind NATs and firewalls.

PEER-TO-PEER NETWORKS

PROBLEMS: CLIENT RESOURCES



The complexity of most of the P2P applications, result in they can not be executed on some types of devices. Especially in those with low resources.

RESEARCH QUESTIONS

RESEARCH QUESTIONS

Is it possible to design a P2P protocol which reduce the communication cost to the maximum?

RESEARCH QUESTIONS

Is it possible to design a P2P protocol which reduce the communication cost to the maximum?

How could we detect and expel (or force to contribute) to Malicious Peers?

RESEARCH QUESTIONS

Is it possible to design a P2P protocol which reduce the communication cost to the maximum?

How could we detect and expel (or force to contribute) to Malicious Peers?

How could we increase the chances two peers perform a successful communication?

RESEARCH QUESTIONS

Is it possible to design a P2P protocol which reduce the communication cost to the maximum?

How could we detect and expel (or force to contribute) to Malicious Peers?

How could we increase the chances two peers perform a successful communication?

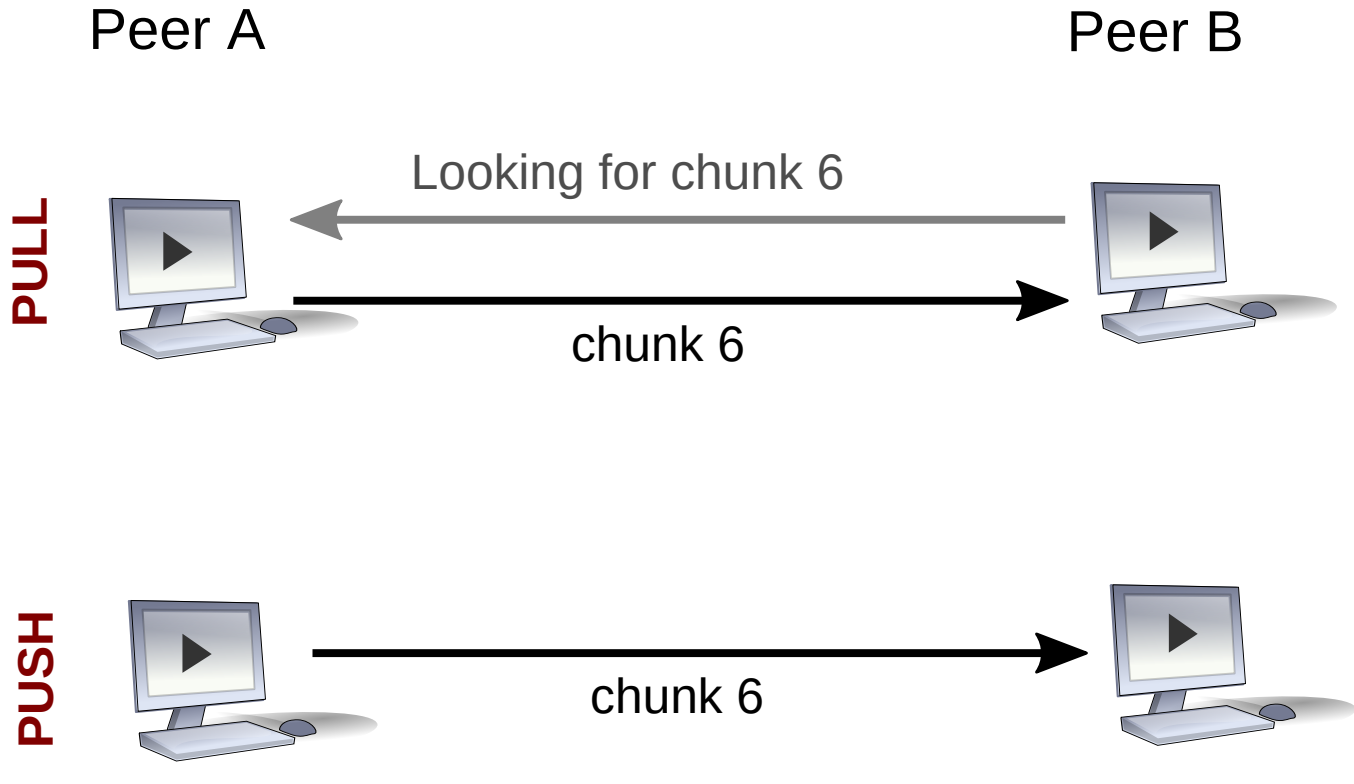
Would our protocol be straighforward enough to be run on low-resource devices?

OUTLINE

- INTRODUCTION
  - Background
  - Research Questions
- **PEER TO PEER STRAIGHFORWARD PROTOCOL**
  - An application-layer protocol that provides real-time broadcasting on the Internet
- SECURITY
  - Pollution Attacks
  - Strategies applied to Traditional Networks
  - Strategies applied to Software Defined Networks
- EFFICIENCY
  - NAT Traversal using Collaborative Port Prediction
  - Runing P2PSP on Embeded Devices
- CONCLUSIONS
  - Key Results and Future Work

PULL-MODEL VS PUSH-MODEL

GOAL: REDUCE LATENCY BY KEEPING IT AS SIMPLE AS POSSIBLE




P2PSP
Peer to Peer Straightforward Protocol

- **P2PSP (PEER TO PEER STRAIGHTFORWARD PROTOCOL) IS AN OPEN APPLICATION-LAYER PROTOCOL FOR THE REAL-TIME STREAMING OF MEDIA CONTENT OVER A PEER-TO-PEER OVERLAY.**

- **P2PSP (PEER TO PEER STRAIGHTFORWARD PROTOCOL) IS AN OPEN APPLICATION-LAYER PROTOCOL FOR THE REAL-TIME STREAMING OF MEDIA CONTENT OVER A PEER-TO-PEER OVERLAY.**
- **P2PSP IS BASED ON A PUSH-BASED FULLY CONNECTED MESH SCHEME WHERE EVERY PEER IS CONNECTED WITH EACH OTHER.**

- P2PSP (PEER TO PEER STRAIGHTFORWARD PROTOCOL) IS AN OPEN APPLICATION-LAYER PROTOCOL FOR THE REAL-TIME STREAMING OF MEDIA CONTENT OVER A PEER-TO-PEER OVERLAY.
- P2PSP IS BASED ON A PUSH-BASED FULLY CONNECTED MESH SCHEME WHERE EVERY PEER IS CONNECTED WITH EACH OTHER.
- AN OPEN-SOURCE (GNU GPL V3) IMPLEMENTATION IS AVAILABLE ON GITHUB.

MODULAR DESIGN

MODULAR DESIGN

LBS (Load Balancing Set): P2PSP supposes that there is a collection of channels that are broadcasted in parallel.

MODULAR DESIGN

LBS (Load Balancing Set): P2PSP supposes that there is a collection of channels that are broadcasted in parallel.
DBS (Data Broadcasting Set): Designed to be efficient in transmitting a data-stream from a splitter node to peers.

MODULAR DESIGN

LBS (Load Balancing Set): P2PSP supposes that there is a collection of channels that are broadcasted in parallel.
DBS (Data Broadcasting Set): Designed to be efficient in transmitting a data-stream from a splitter node to peers.
IMS (IP MUlticast Set): Peers in the same local network communicate using IPM group address and port, if available.

MODULAR DESIGN

LBS (Load Balancing Set): P2PSP supposes that there is a collection of channels that are broadcasted in parallel.
DBS (Data Broadcasting Set): Designed to be efficient in transmitting a data-stream from a splitter node to peers.
IMS (IP MUlticast Set): Peers in the same local network communicate using IPM group address and port, if available.
MRS (Massively-lost chunk Recovery Set): A massively-lost chunk occurs when a chunk is lost in its way from the splitter to a peer.

MODULAR DESIGN

LBS (Load Balancing Set): P2PSP supposes that there is a collection of channels that are broadcasted in parallel.
DBS (Data Broadcasting Set): Designed to be efficient in transmitting a data-stream from a splitter node to peers.
IMS (IP MUlticast Set): Peers in the same local network communicate using IPM group address and port, if available.
MRS (Massively-lost chunk Recovery Set): A massively-lost chunk occurs when a chunk is lost in its way from the splitter to a peer.
ACS (Adaptive Capacity Set): It relaxes the peer's upload requirements imposed by DBS.

MODULAR DESIGN

LBS (Load Balancing Set): P2PSP supposes that there is a collection of channels that are broadcasted in parallel.
DBS (Data Broadcasting Set): Designed to be efficient in transmitting a data-stream from a splitter node to peers.
IMS (IP MUlticast Set): Peers in the same local network communicate using IPM group address and port, if available.
MRS (Massively-lost chunk Recovery Set): A massively-lost chunk occurs when a chunk is lost in its way from the splitter to a peer.
ACS (Adaptive Capacity Set): It relaxes the peer's upload requirements imposed by DBS.
NTS (NAT Traversal Set): It provides peer connectivity for some NAT configurations where DBS can not establish a direct peer communication.

MODULAR DESIGN

LBS (Load Balancing Set): P2PSP supposes that there is a collection of channels that are broadcasted in parallel.
DBS (Data Broadcasting Set): Designed to be efficient in transmitting a data-stream from a splitter node to peers.
IMS (IP MUlticast Set): Peers in the same local network communicate using IPM group address and port, if available.
MRS (Massively-lost chunk Recovery Set): A massively-lost chunk occurs when a chunk is lost in its way from the splitter to a peer.
ACS (Adaptive Capacity Set): It relaxes the peer's upload requirements imposed by DBS.
NTS (NAT Traversal Set): It provides peer connectivity for some NAT configurations where DBS can not establish a direct peer communication.
MCS (Multi-Channel Set): Scalable Video Coding compatibility.

MODULAR DESIGN

LBS (Load Balancing Set): P2PSP supposes that there is a collection of channels that are broadcasted in parallel.
DBS (Data Broadcasting Set): Designed to be efficient in transmitting a data-stream from a splitter node to peers.
IMS (IP MUlticast Set): Peers in the same local network communicate using IPM group address and port, if available.
MRS (Massively-lost chunk Recovery Set): A massively-lost chunk occurs when a chunk is lost in its way from the splitter to a peer.
ACS (Adaptive Capacity Set): It relaxes the peer's upload requirements imposed by DBS.
NTS (NAT Traversal Set): It provides peer connectivity for some NAT configurations where DBS can not establish a direct peer communication.
MCS (Multi-Channel Set): Scalable Video Coding compatibility.
CIS (Content Integrity Set): The main goal of this set of rules is to face pollution attacks.

MODULAR DESIGN

LBS (Load Balancing Set): P2PSP supposes that there is a collection of channels that are broadcasted in parallel.

DBS (Data Broadcasting Set): Designed to be efficient in transmitting a data-stream from a splitter node to peers.

IMS (IP MUlticast Set): Peers in the same local network communicate using IPM group address and port, if available.

MRS (Massively-lost chunk Recovery Set): A massively-lost chunk occurs when a chunk is lost in its way from the splitter to a peer.

ACS (Adaptive Capacity Set): It relaxes the peer's upload requirements imposed by DBS.

NTS (NAT Traversal Set): It provides peer connectivity for some NAT configurations where DBS can not establish a direct peer communication.

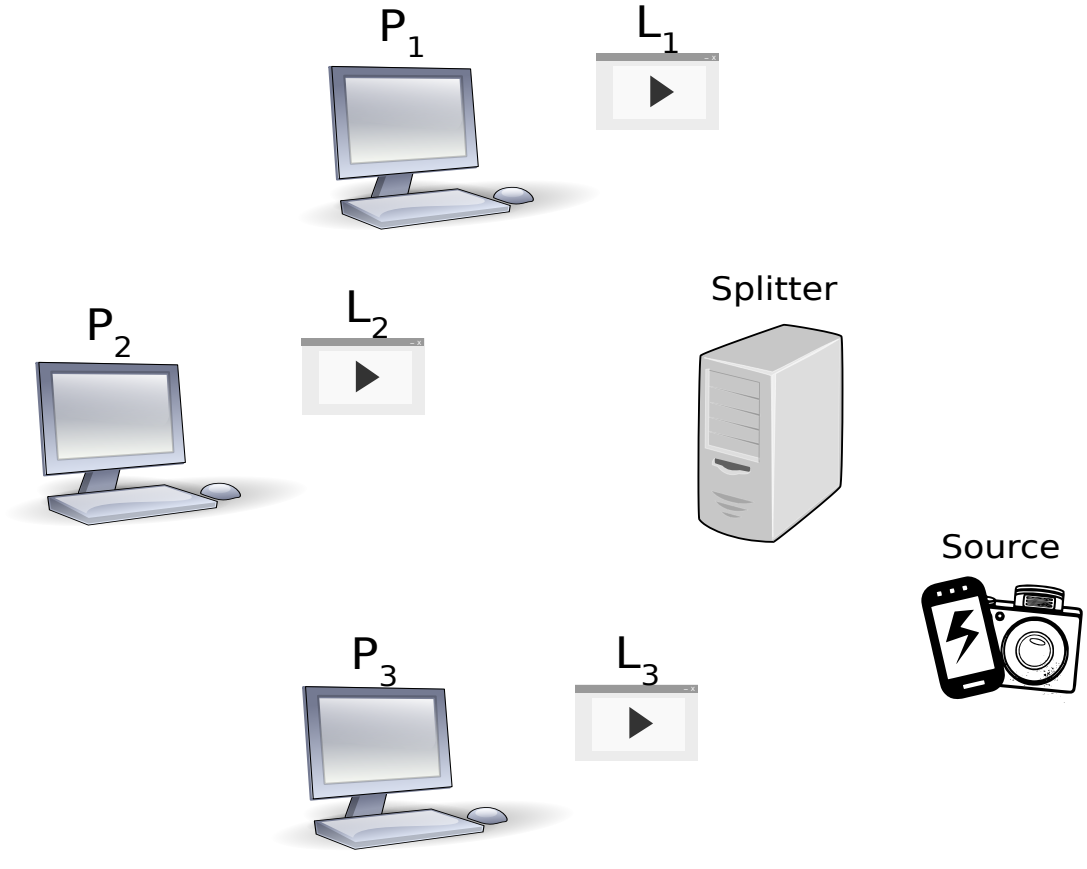
MCS (Multi-Channel Set): Scalable Video Coding compatibility.

CIS (Content Integrity Set): The main goal of this set of rules is to face pollution attacks.

HOW DOES A P2PSP SYSTEM WORK?

An open-source implementation is available on GitHub

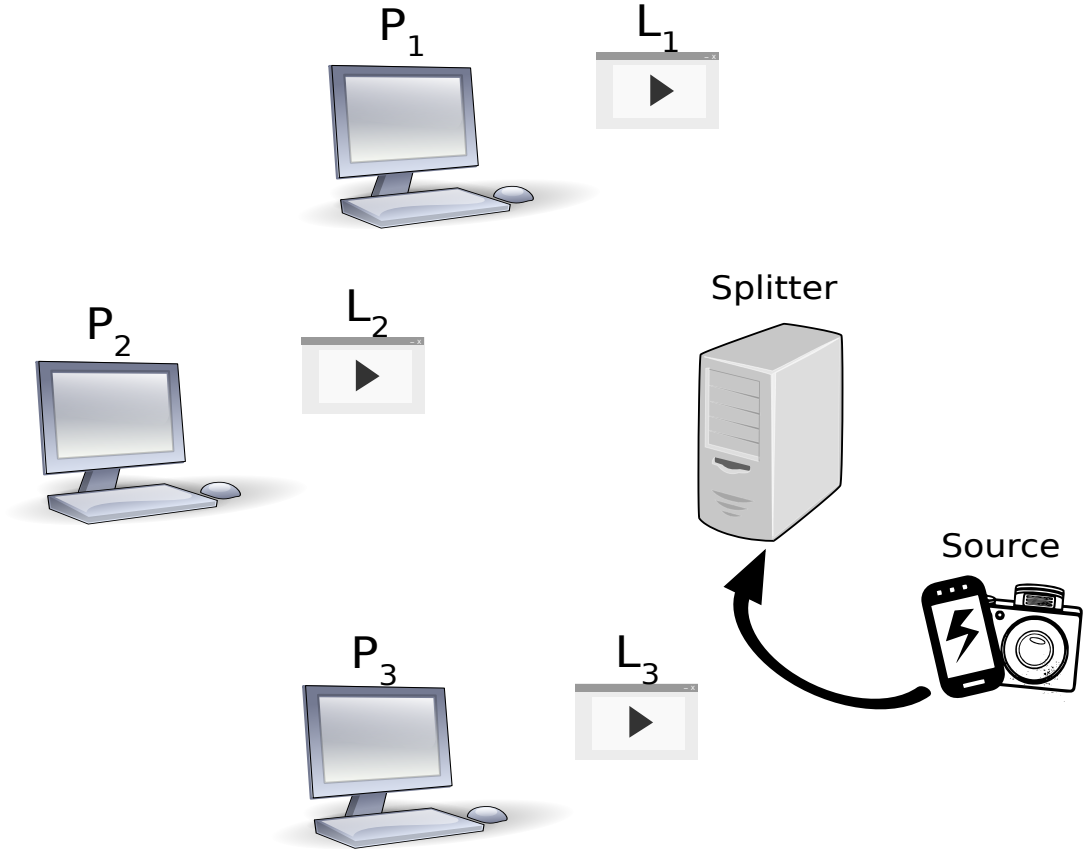
A P2PSP TEAM



HOW DOES A P2PSP SYSTEM WORK?

An open-source implementation is available on GitHub

A P2PSP TEAM

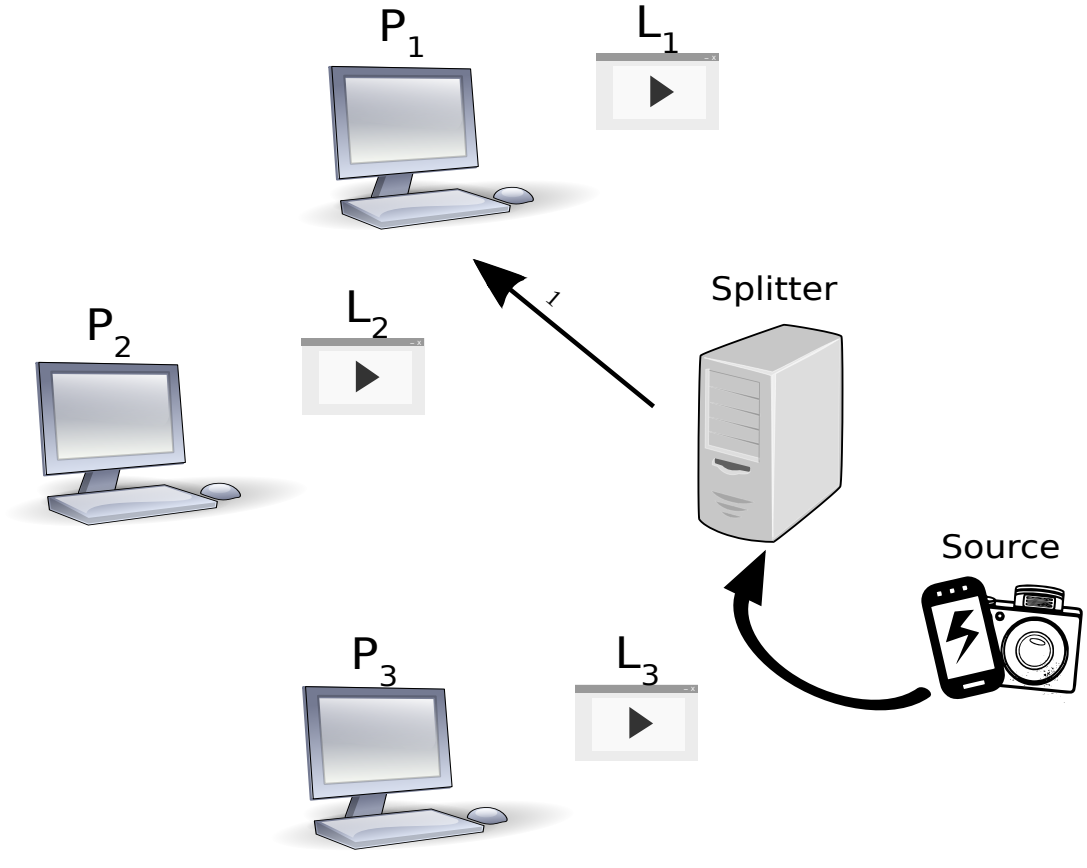


1. THE VIDEO IS SENT IN REAL TIME TO THE SPLITTER.

HOW DOES A P2PSP SYSTEM WORK?

An open-source implementation is available on GitHub

A P2PSP TEAM



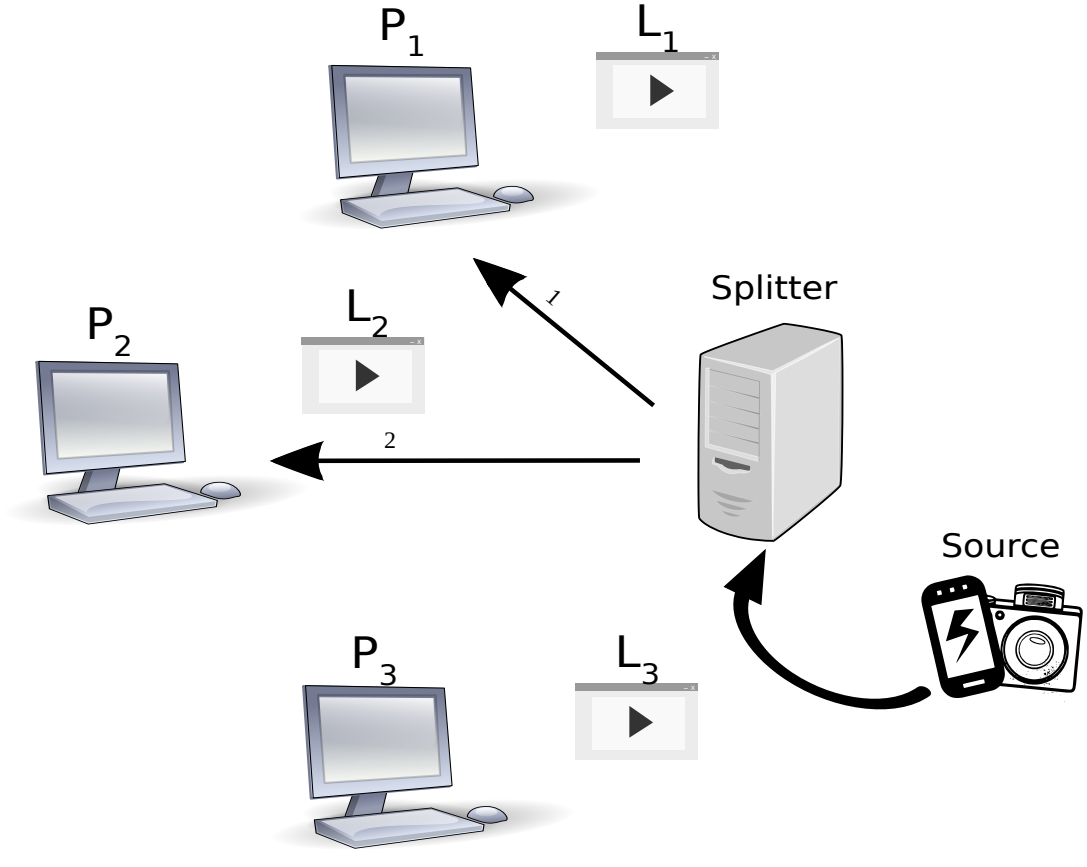
1. THE VIDEO IS SENT IN REAL TIME TO THE SPLITTER.

2. THE SPLITTER DIVIDES THE STREAM IN SEVERAL CHUNKS AND EVERY CHUNK IS SENT TO ONE DIFFERENT PEER.

HOW DOES A P2PSP SYSTEM WORK?

An open-source implementation is available on GitHub

A P2PSP TEAM



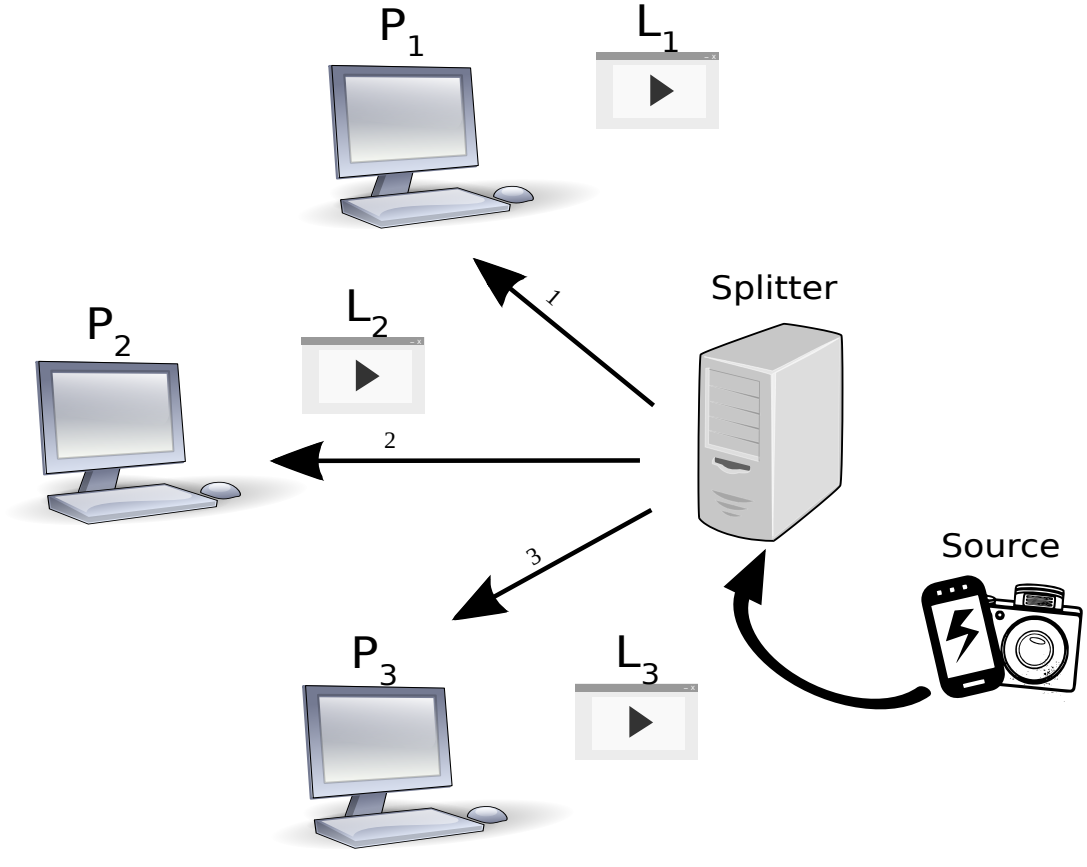
1. THE VIDEO IS SENT IN REAL TIME TO THE SPLITTER.

2. THE SPLITTER DIVIDES THE STREAM IN SEVERAL CHUNKS AND EVERY CHUNK IS SENT TO ONE DIFFERENT PEER.

HOW DOES A P2PSP SYSTEM WORK?

An open-source implementation is available on GitHub

A P2PSP TEAM



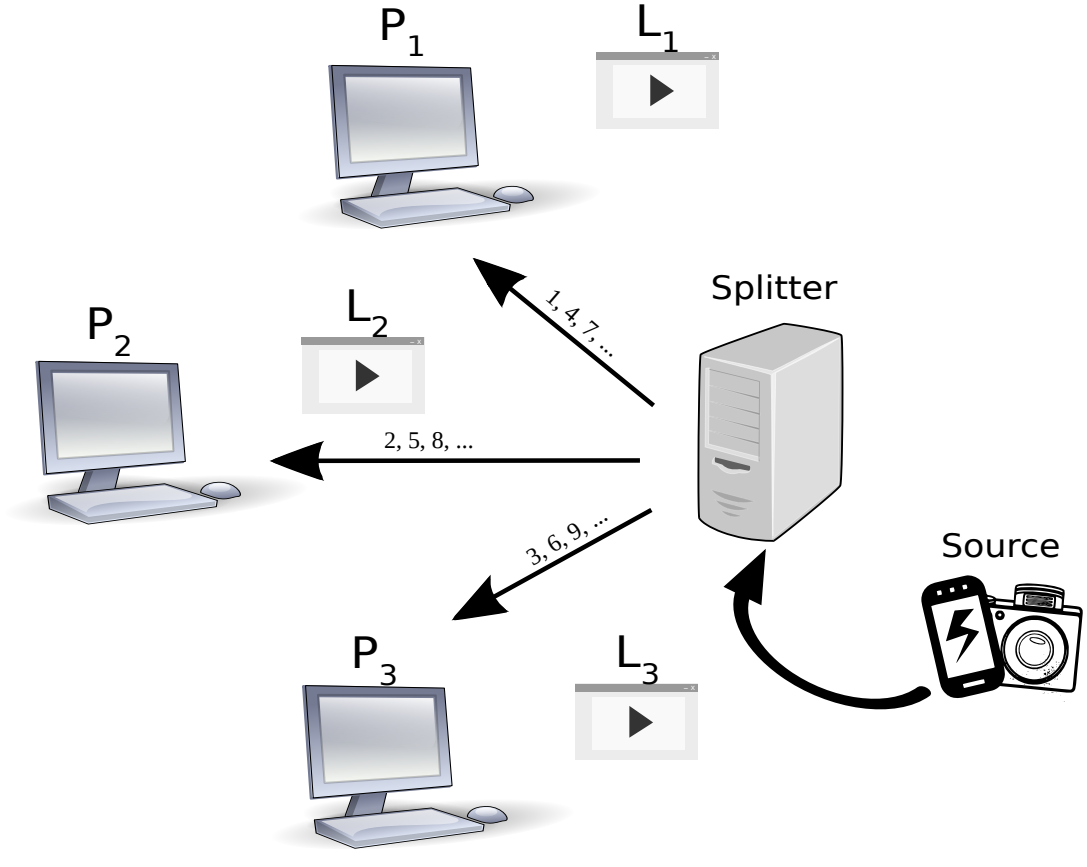
1. THE VIDEO IS SENT IN REAL TIME TO THE SPLITTER.

2. THE SPLITTER DIVIDES THE STREAM IN SEVERAL CHUNKS AND EVERY CHUNK IS SENT TO ONE DIFFERENT PEER.

HOW DOES A P2PSP SYSTEM WORK?

An open-source implementation is available on GitHub

A P2PSP TEAM



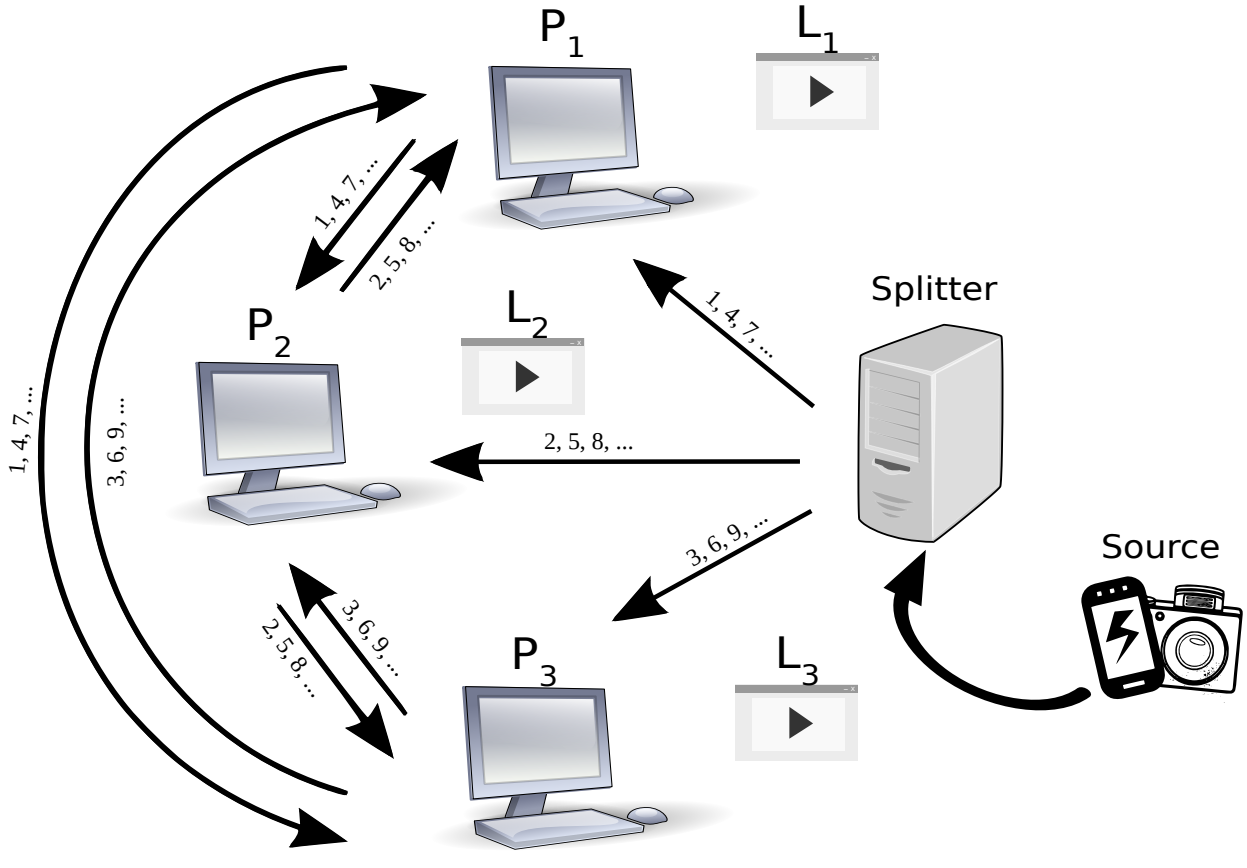
1. THE VIDEO IS SENT IN REAL TIME TO THE SPLITTER.

2. THE SPLITTER DIVIDES THE STREAM IN SEVERAL CHUNKS AND EVERY CHUNK IS SENT TO ONE DIFFERENT PEER.

HOW DOES A P2PSP SYSTEM WORK?

An open-source implementation is available on GitHub

A P2PSP TEAM



1. THE VIDEO IS SENT IN REAL TIME TO THE SPLITTER.

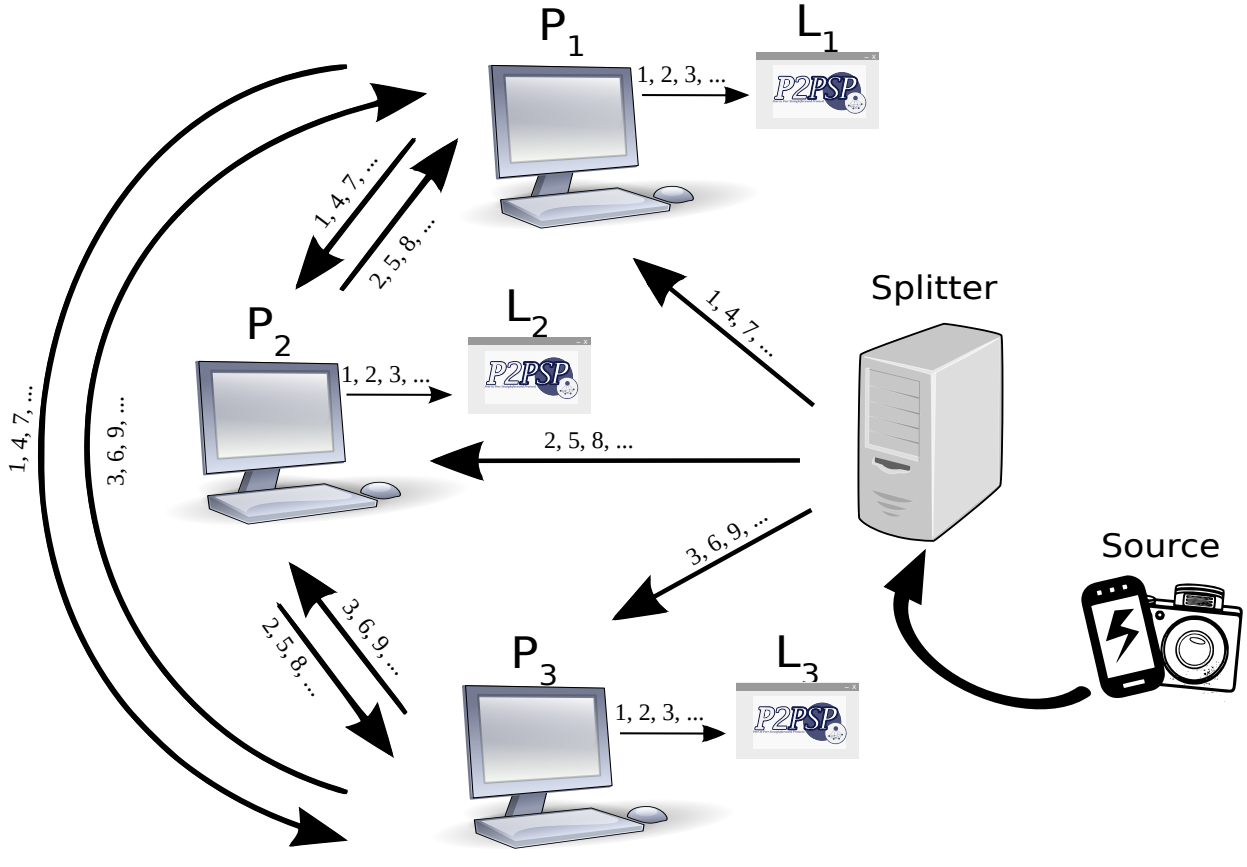
2. THE SPLITTER DIVIDES THE STREAM IN SEVERAL CHUNKS AND EVERY CHUNK IS SENT TO ONE DIFFERENT PEER.

3. EACH PEER SENDS ITS CHUNKS TO EACH OTHER IN ORDER TO ENSURE THAT EVERYONE HAS THE WHOLE STREAM.

HOW DOES A P2PSP SYSTEM WORK?

An open-source implementation is available on GitHub

A P2PSP TEAM



1. THE VIDEO IS SENT IN REAL TIME TO THE SPLITTER.

2. THE SPLITTER DIVIDES THE STREAM IN SEVERAL CHUNKS AND EVERY CHUNK IS SENT TO ONE DIFFERENT PEER.

3. EACH PEER SENDS ITS CHUNKS TO EACH OTHER IN ORDER TO ENSURE THAT EVERYONE HAS THE WHOLE STREAM.

4. PEERS SEND THE STREAM TO THE PLAYER.

OUTLINE

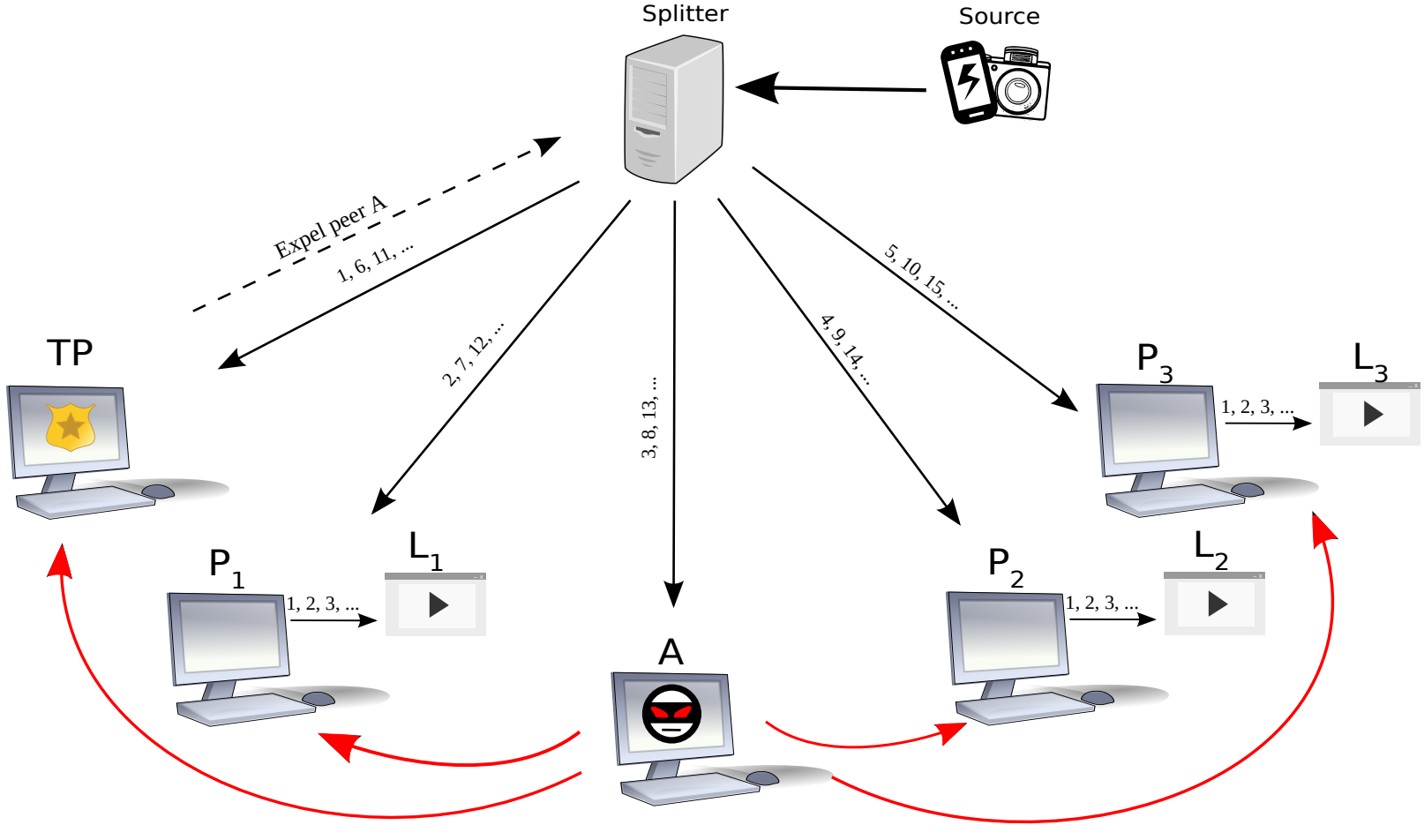
- INTRODUCTION
  - Background
  - Research Questions
- PEER TO PEER STRAIGHFORWARD PROTOCOL
  - An application-layer protocol that provides real-time broadcasting on the Internet
- **SECURITY**
  - Pollution Attacks
  - Strategies applied to Traditional Networks
  - Strategies applied to Software Defined Networks
- EFFICIENCY
  - NAT Traversal using Collaborative Port Prediction
  - Runing P2PSP on Embeded Devices
- CONCLUSIONS
  - Key Results and Future Work

POLLUTION ATTACKS

Pollution attacks consist of a peer or a set of peers modifying the content of the stream.
Can be done in different ways.

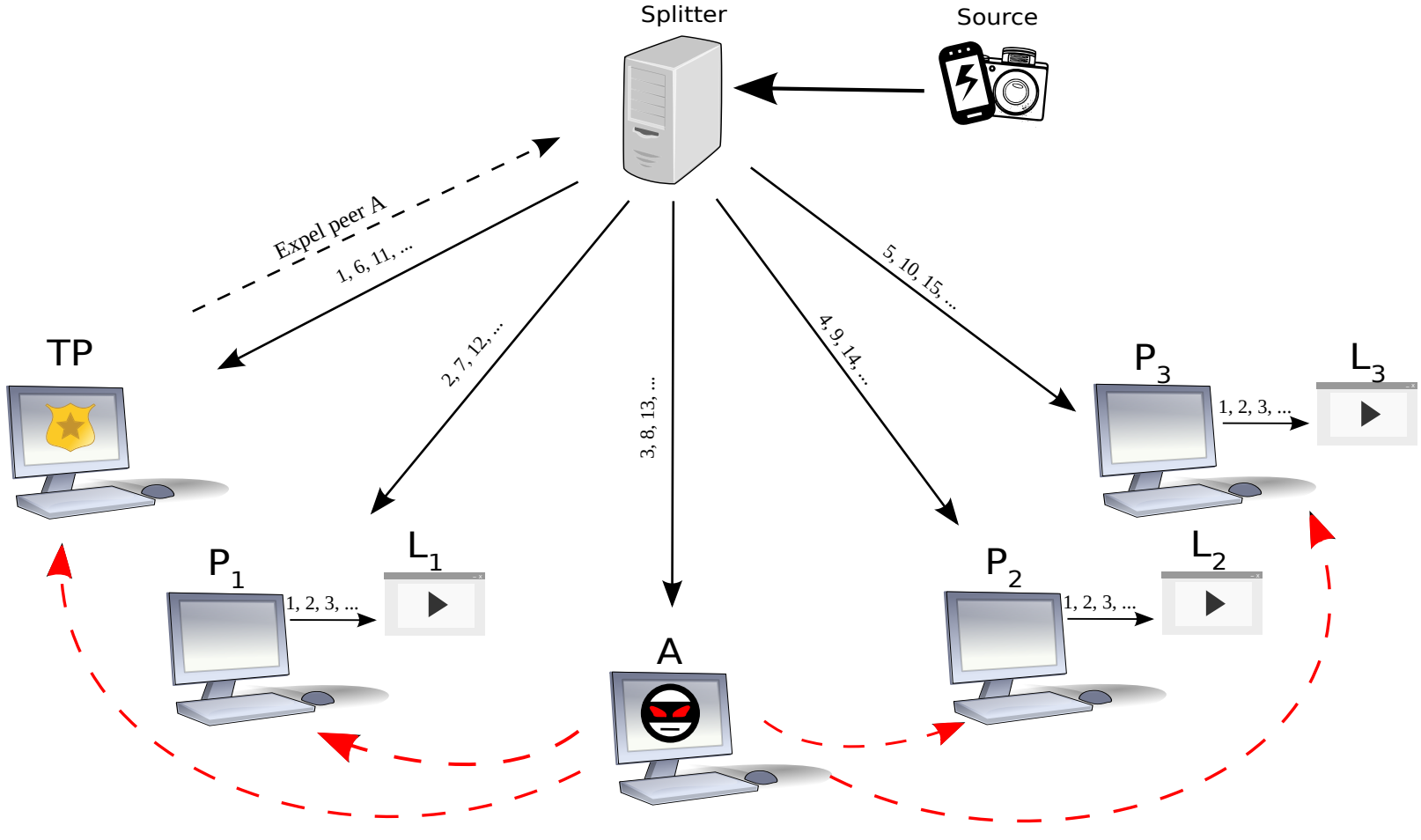
POLLUTION ATTACKS

Persistent attack: an attacker poisons every chunk received from the splitter and sends them to the entire team.



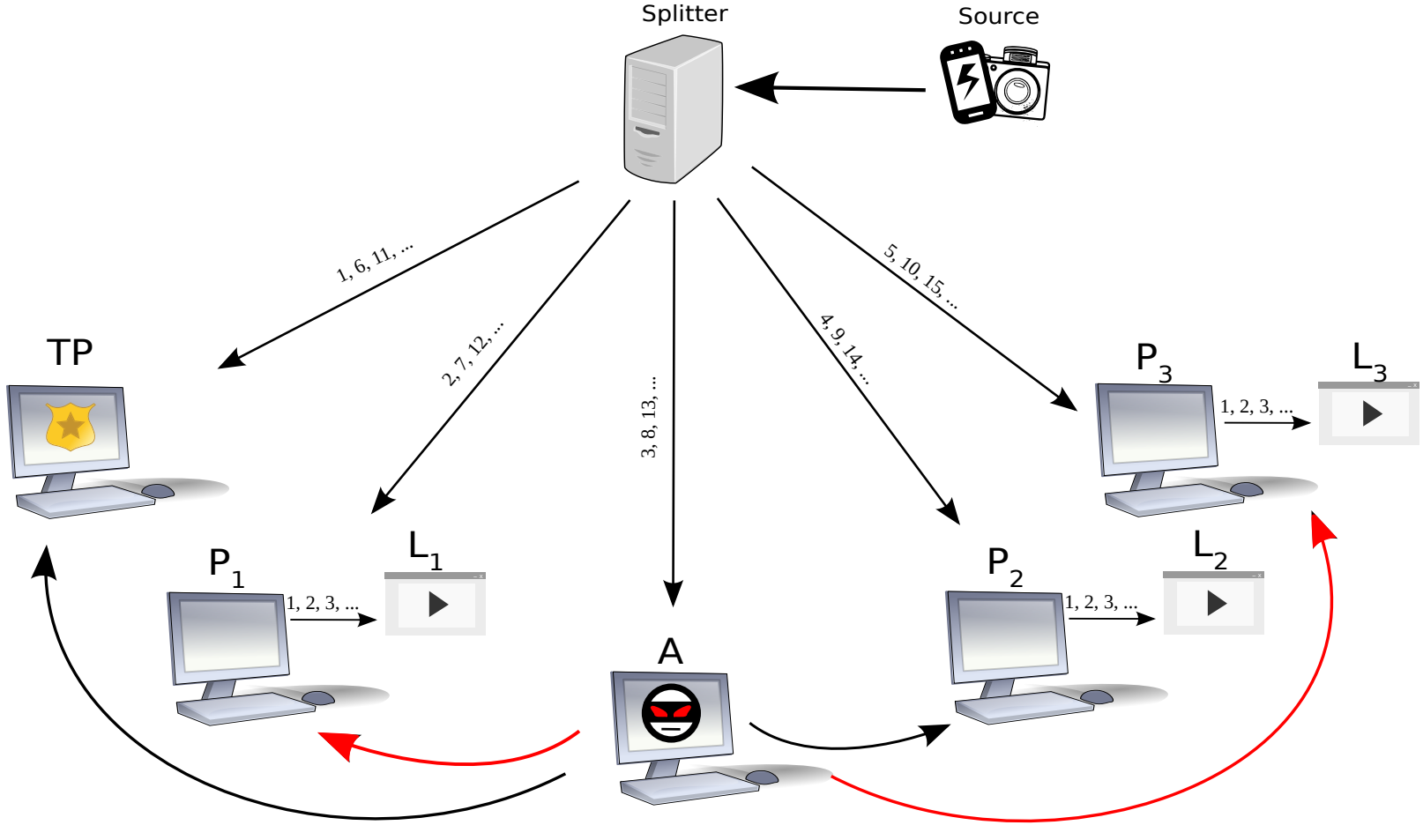
POLLUTION ATTACKS

On-Off attack: the attacker only poisons some chunks but not others.



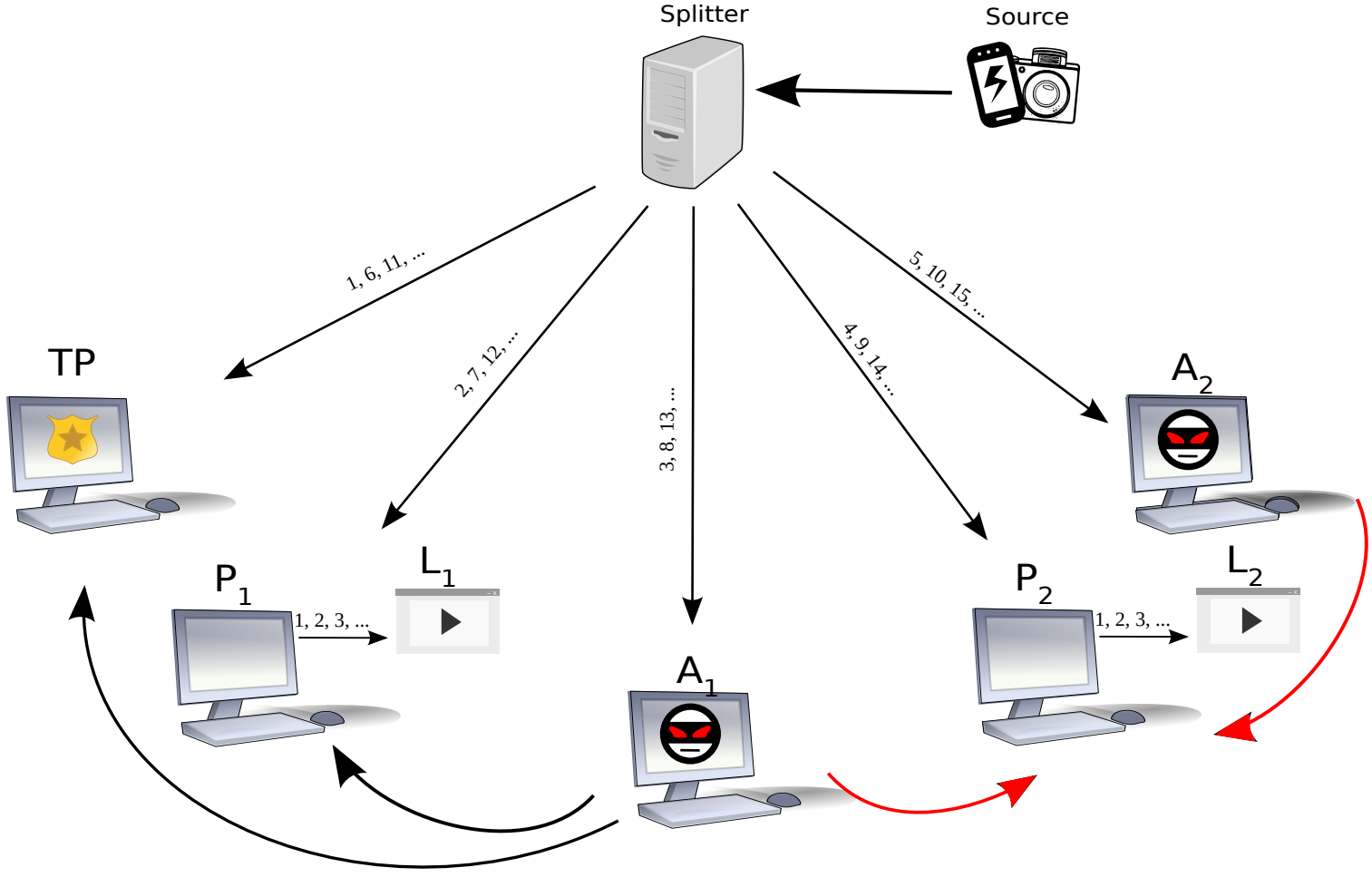
POLLUTION ATTACKS

Selective attack: poisoning chunks intended for only one peer or a small subset of peers.



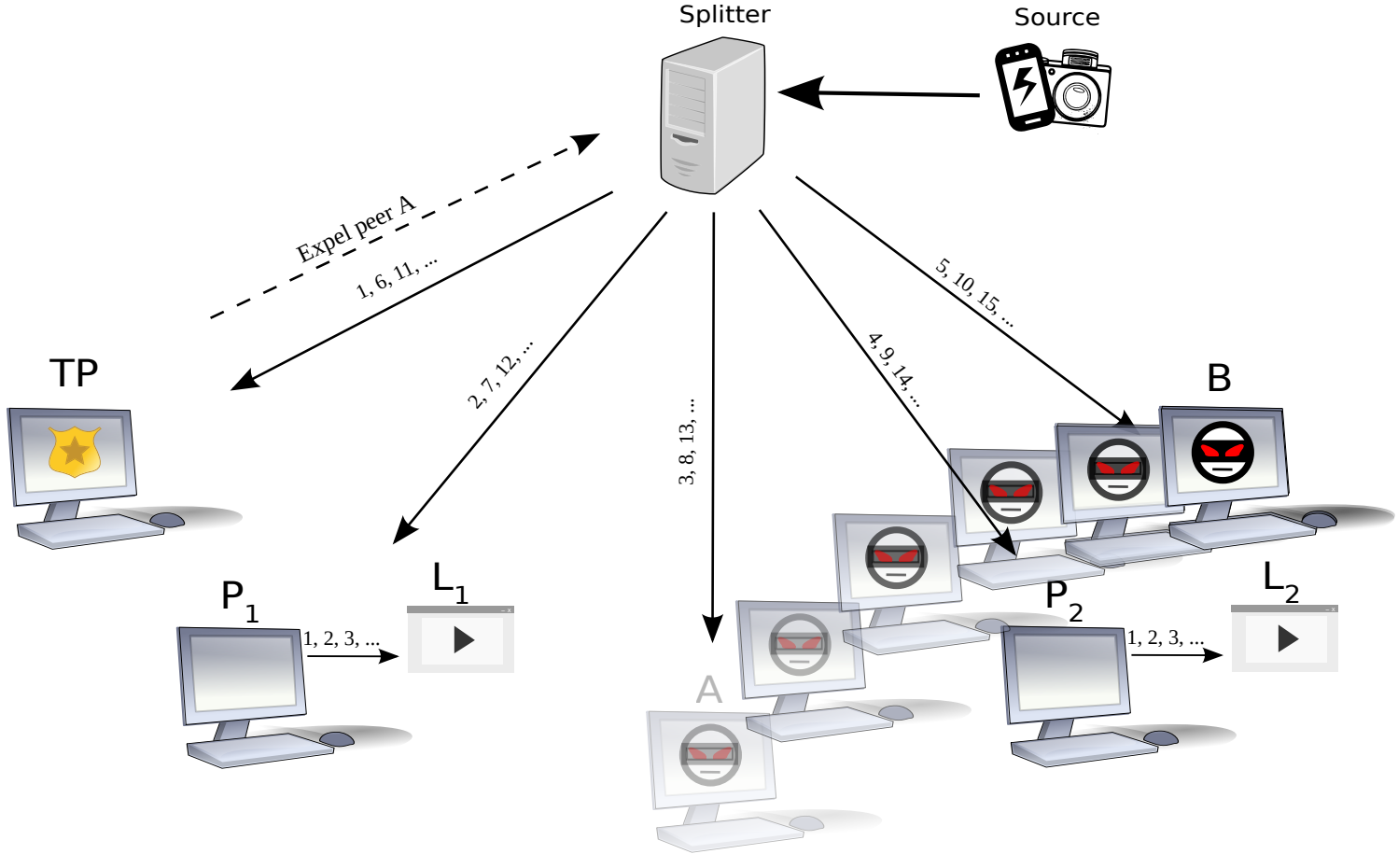
POLLUTION ATTACKS

Collaborative attack: several attackers may collaborate to produce Selective and On-off attacks on a large set of peers.



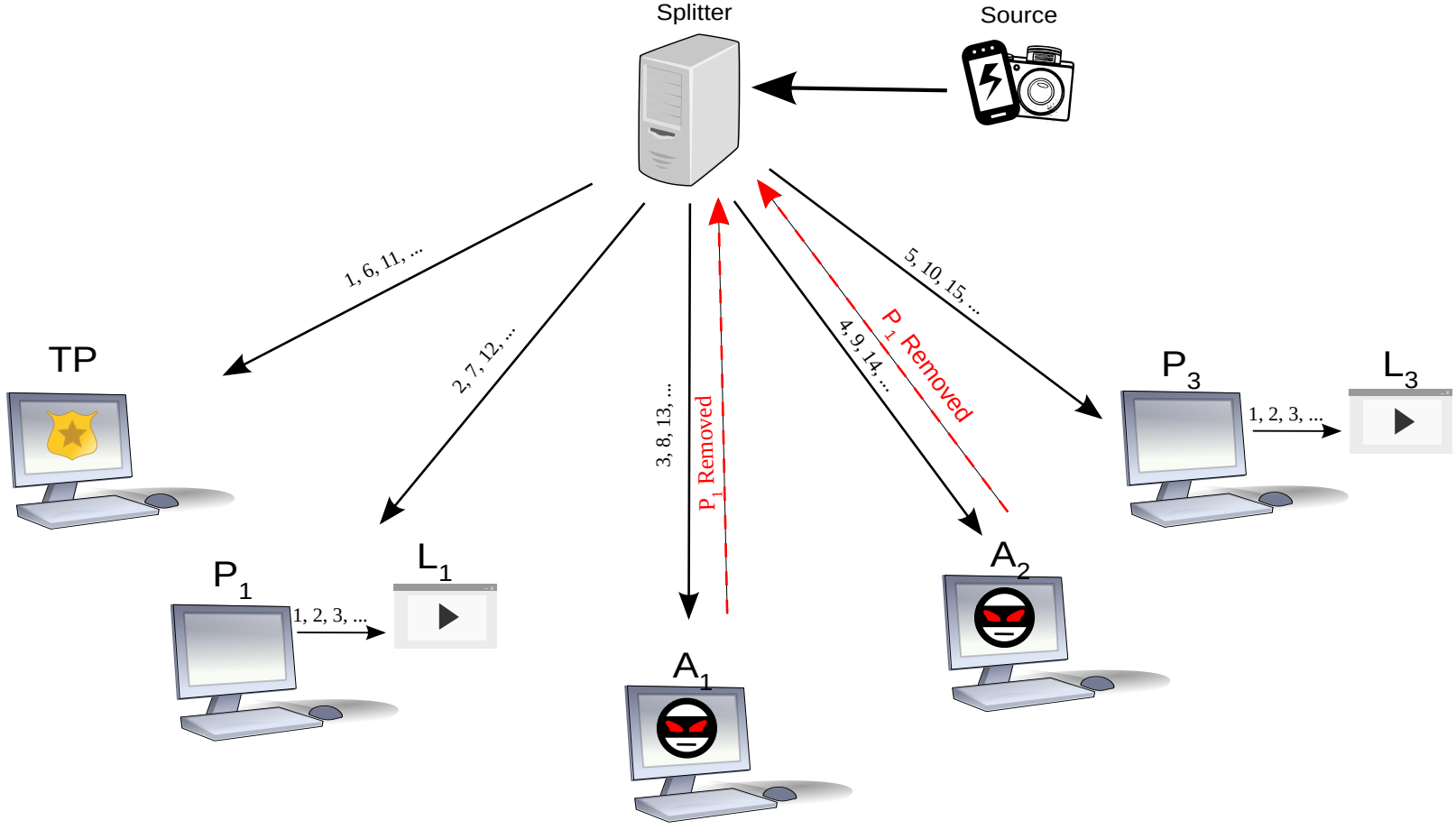
POLLUTION ATTACKS

Hand-wash attack: leaving the team and returning to continue the attack with another alias.



POLLUTION ATTACKS

Bad-mouth attack: blaming other regular peers of sending poisoned chunks or not sending chunks.

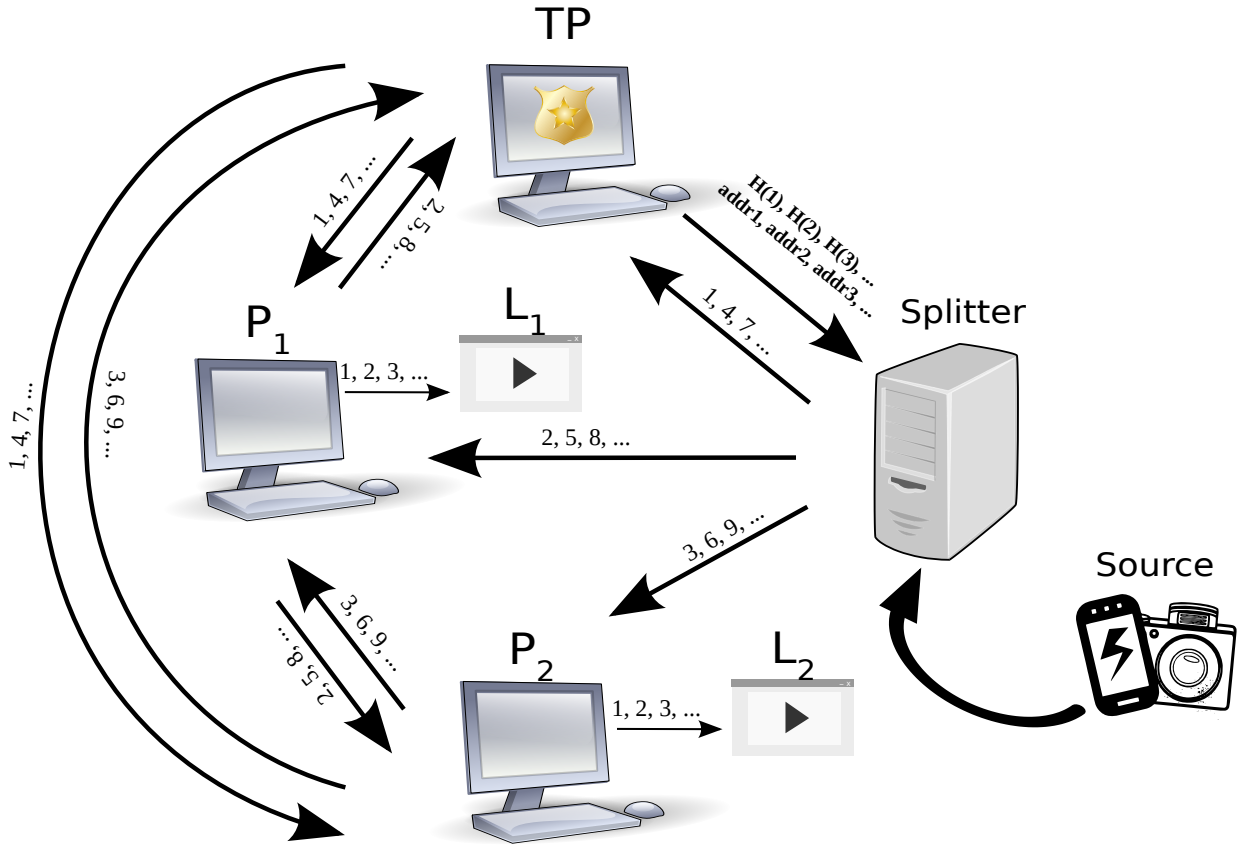


OUTLINE

- INTRODUCTION
  - Background
  - Research Questions
- PEER TO PEER STRAIGHFORWARD PROTOCOL
  - An application-layer protocol that provides real-time broadcasting on the Internet
- **SECURITY**
  - Pollution Attacks
  - Strategies applied to Traditional Networks
  - Strategies applied to Software Defined Networks
- EFFICIENCY
  - NAT Traversal using Collaborative Port Prediction
  - Runing P2PSP on Embeded Devices
- CONCLUSIONS
  - Key Results and Future Work

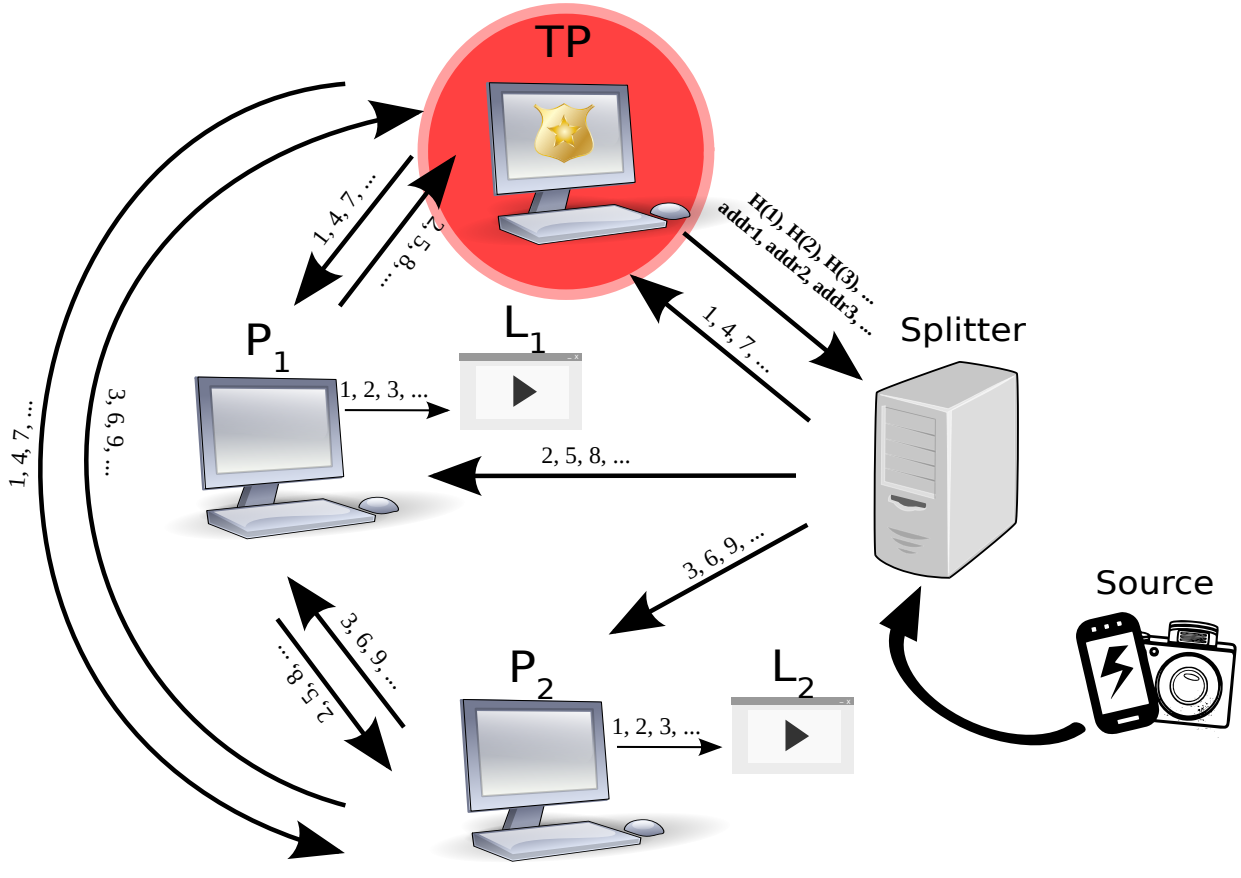
STRATEGY BASED ON TRUSTED PEERS

HOW DOES IT WORK?



STRATEGY BASED ON TRUSTED PEERS

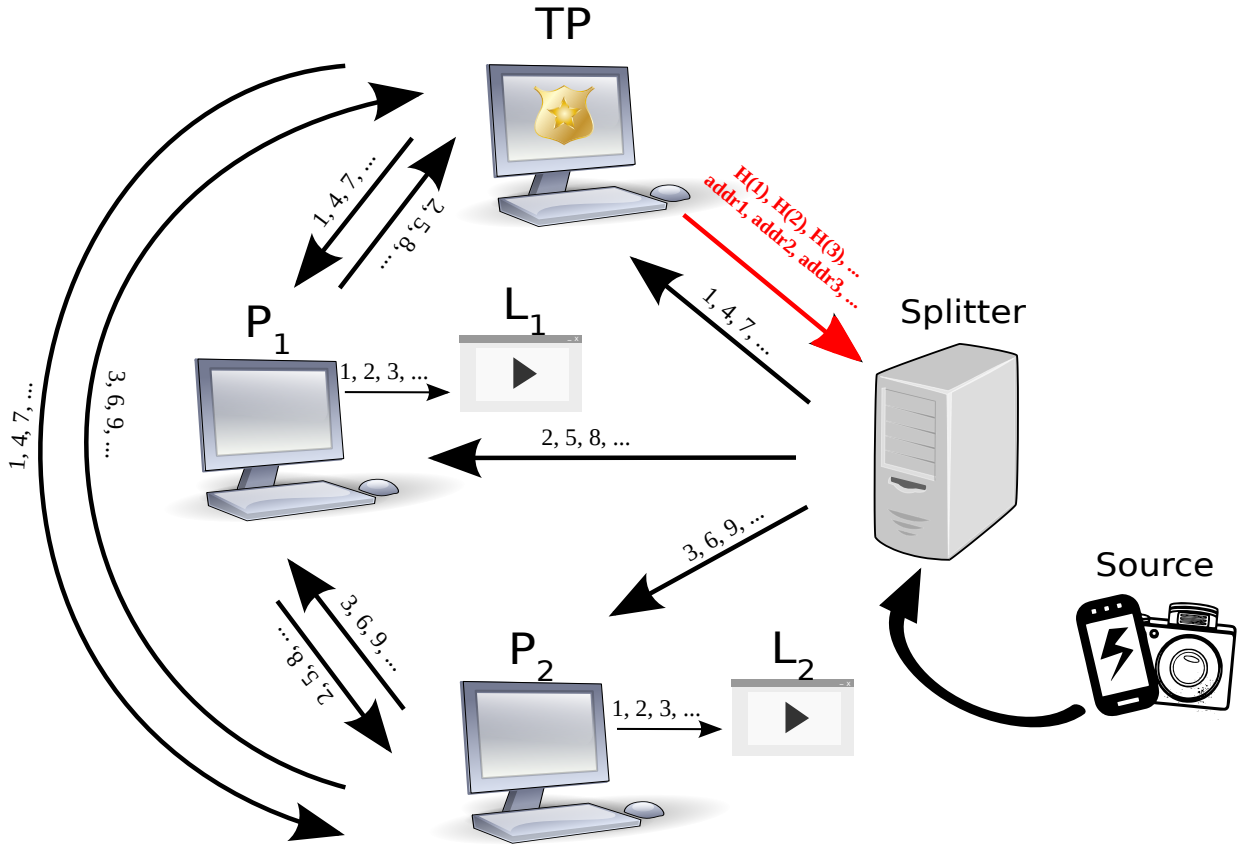
HOW DOES IT WORK?



1. ONLY THE SPLITTER KNOWS WHO THE TPS IN THE TEAM ARE.

STRATEGY BASED ON TRUSTED PEERS

HOW DOES IT WORK?

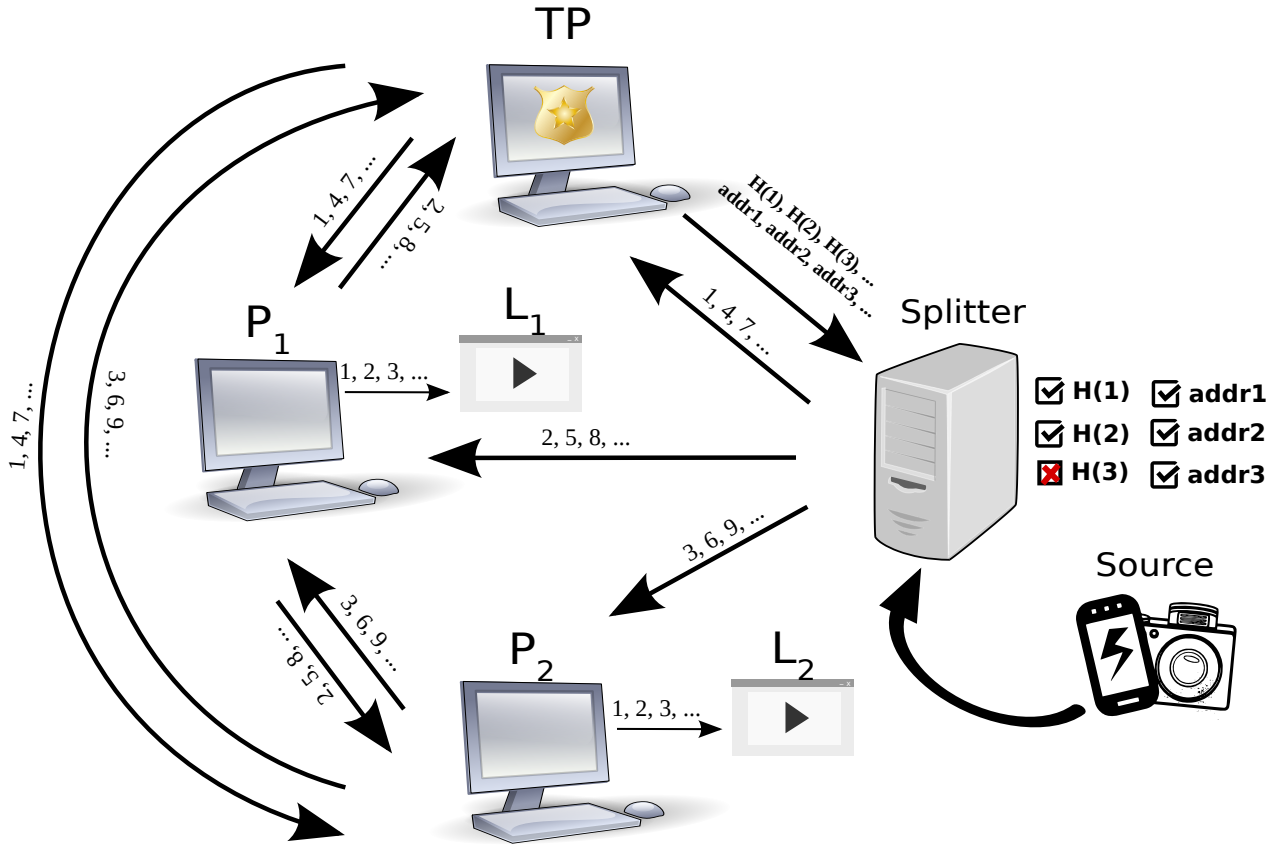


1. ONLY THE SPLITTER KNOWS WHO THE TPS IN THE TEAM ARE.

2. EACH TP CREATES A HASH FOR EACH CHUNK, INCLUDING THE CHUNK NUMBER AND THE ENDPOINT OF THE SOURCE AND SENDS IT TO THE SPLITTER.

STRATEGY BASED ON TRUSTED PEERS

HOW DOES IT WORK?



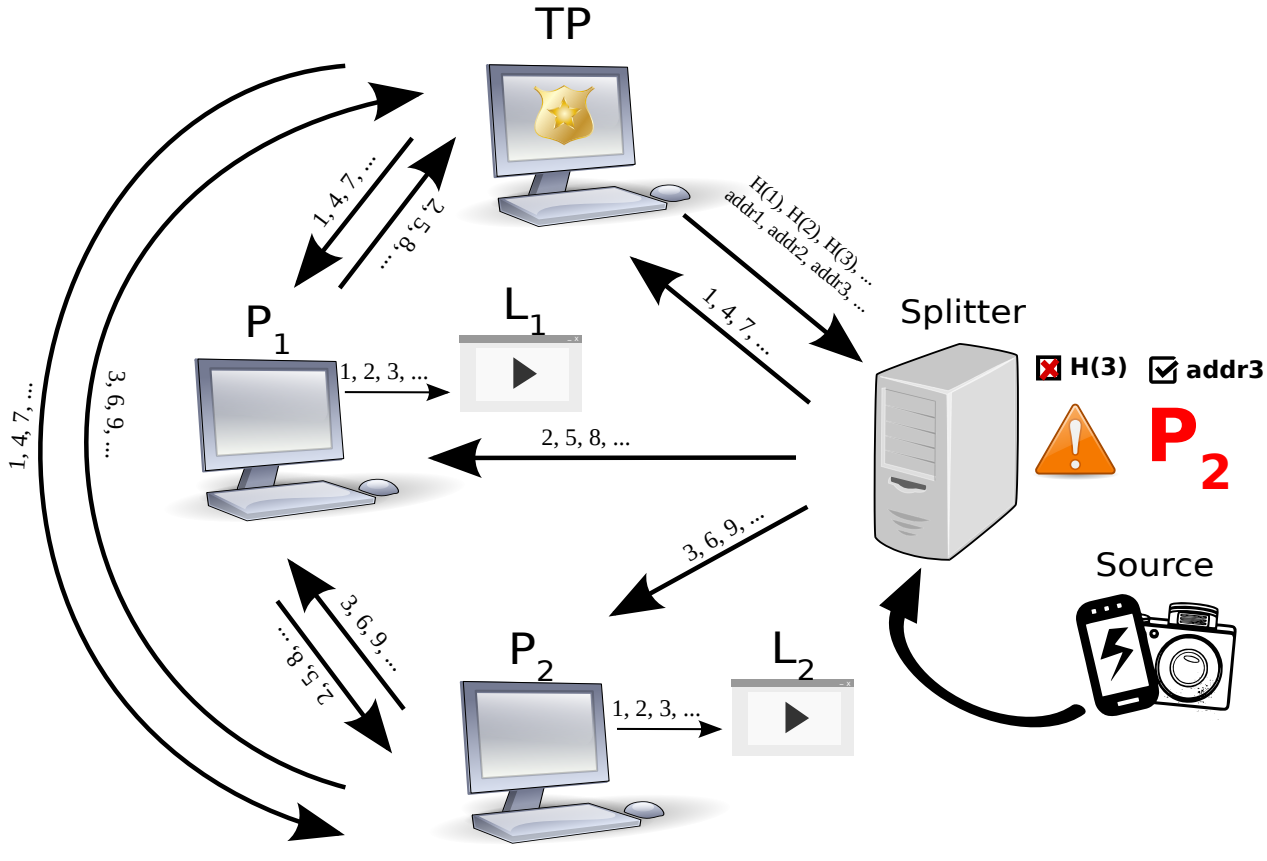
1. ONLY THE SPLITTER KNOWS WHO THE TPS IN THE TEAM ARE.

2. EACH TP CREATES A HASH FOR EACH CHUNK, INCLUDING THE CHUNK NUMBER AND THE ENDPOINT OF THE SOURCE AND SENDS IT TO THE SPLITTER.

3. THE SPLITTER CHECKS WHETHER THE CHUNKS HAVE BEEN ALTERED.

STRATEGY BASED ON TRUSTED PEERS

HOW DOES IT WORK?



1. ONLY THE SPLITTER KNOWS WHO THE TPS IN THE TEAM ARE.

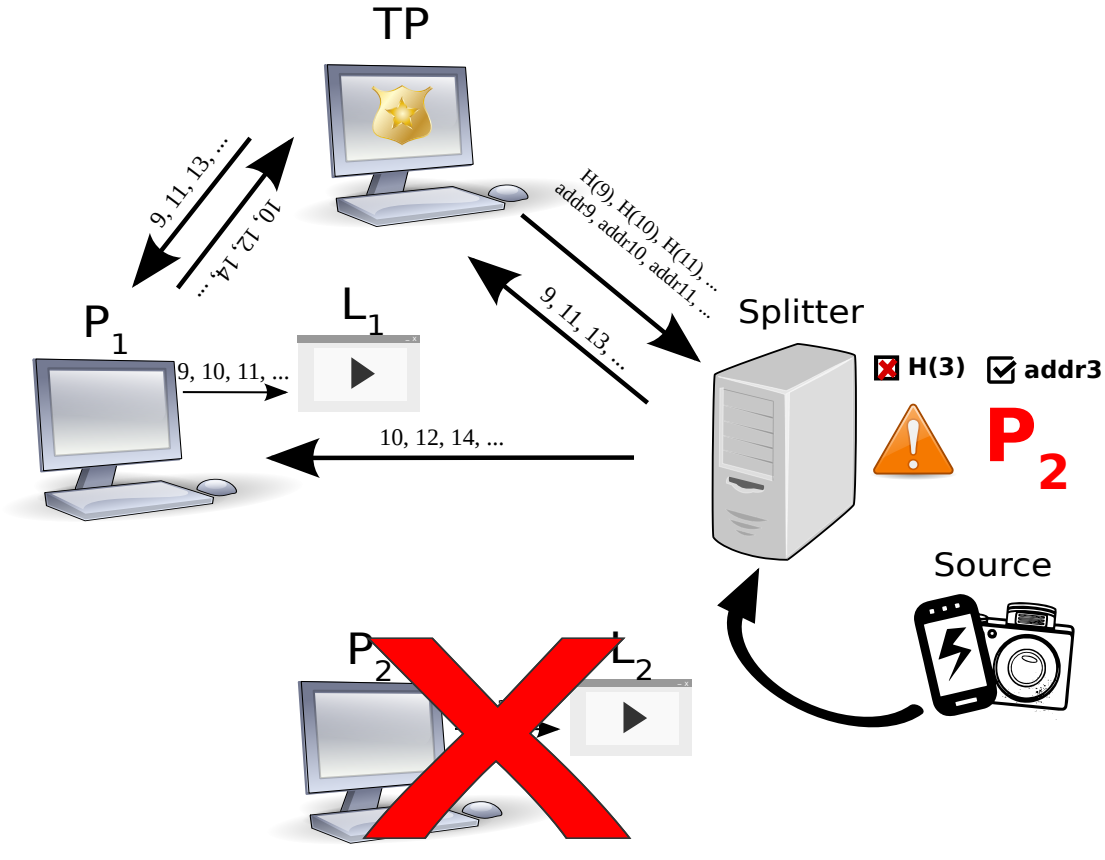
2. EACH TP CREATES A HASH FOR EACH CHUNK, INCLUDING THE CHUNK NUMBER AND THE ENDPOINT OF THE SOURCE AND SENDS IT TO THE SPLITTER.

3. THE SPLITTER CHECKS WHETHER THE CHUNKS HAVE BEEN ALTERED.

4. THE SPLITTER KNOWS THE PEER IN CHARGE OF RELAYING A GIVEN CHUNK.

STRATEGY BASED ON TRUSTED PEERS

HOW DOES IT WORK?



1. ONLY THE SPLITTER KNOWS WHO THE TPS IN THE TEAM ARE.
2. EACH TP CREATES A HASH FOR EACH CHUNK, INCLUDING THE CHUNK NUMBER AND THE ENDPOINT OF THE SOURCE AND SENDS IT TO THE SPLITTER.
3. THE SPLITTER CHECKS WHETHER THE CHUNKS HAVE BEEN ALTERED.
4. THE SPLITTER KNOWS THE PEER IN CHARGE OF RELAYING A GIVEN CHUNK.
5. THE ATTACKER IS EXPELLED FROM THE TEAM.

STRATEGY BASED ON TRUSTED PEERS

PROBLEMS:

STRATEGY BASED ON TRUSTED PEERS

PROBLEMS:

- Peers don't know if they are being poisoned.

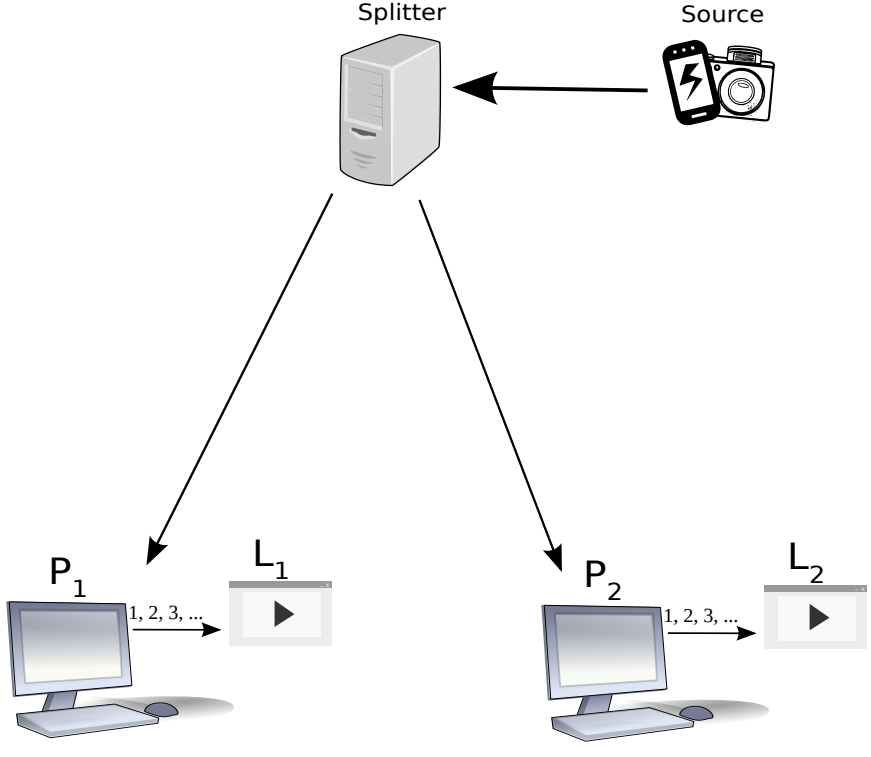
STRATEGY BASED ON TRUSTED PEERS

PROBLEMS:

- Peers don't know if they are being poisoned.
- If an attacker knows who the trusted peers are the system is completely vulnerable to **Selective Attacks**.

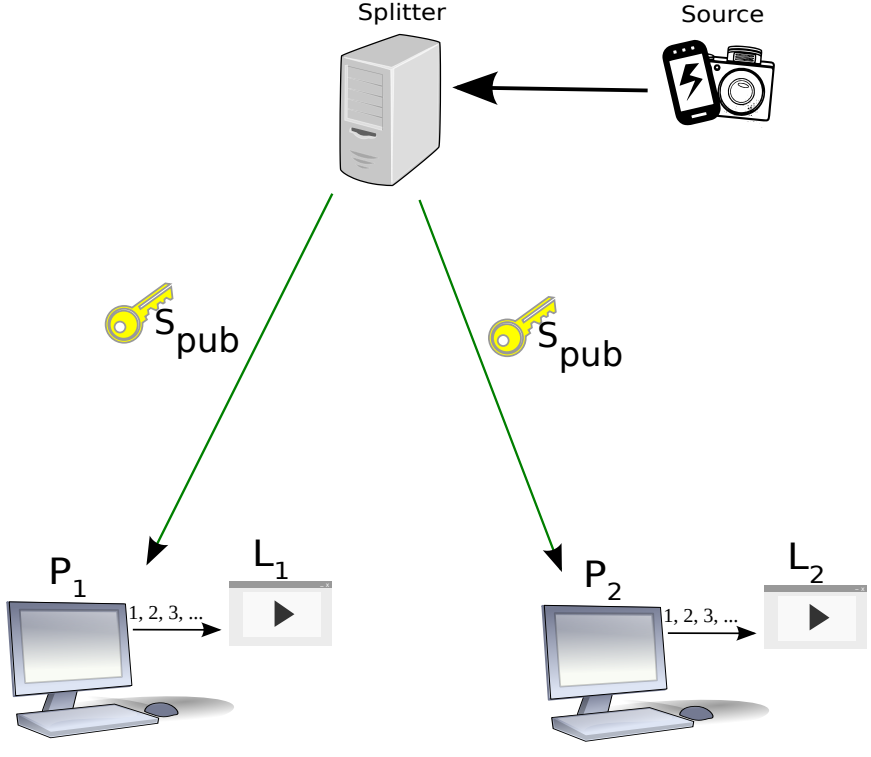
STRATEGY BASED ON TRUSTED PEERS AND DIGITAL SIGNATURES

IT HAS BEEN DESIGNED TO MITIGATE THE SELECTIVE ATTACK AND TO IDENTIFY POISONED CHUNKS BY USING DIGITAL SIGNATURES. THE BEHAVIOR RULES ARE:



STRATEGY BASED ON TRUSTED PEERS AND DIGITAL SIGNATURES

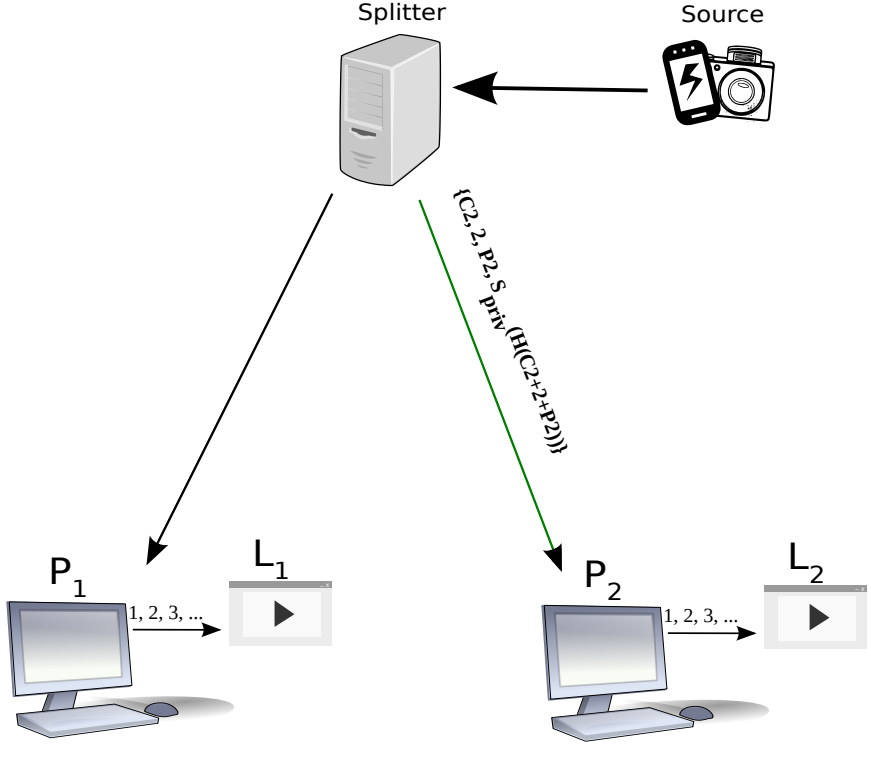
IT HAS BEEN DESIGNED TO MITIGATE THE SELECTIVE ATTACK AND TO IDENTIFY POISONED CHUNKS BY USING DIGITAL SIGNATURES. THE BEHAVIOR RULES ARE:



1.WHEN PEERS JOIN THE TEAM THEY RECEIVE THE PUBLIC KEY OF THE SPLITTER.

STRATEGY BASED ON TRUSTED PEERS AND DIGITAL SIGNATURES

IT HAS BEEN DESIGNED TO MITIGATE THE SELECTIVE ATTACK AND TO IDENTIFY POISONED CHUNKS BY USING DIGITAL SIGNATURES. THE BEHAVIOR RULES ARE:



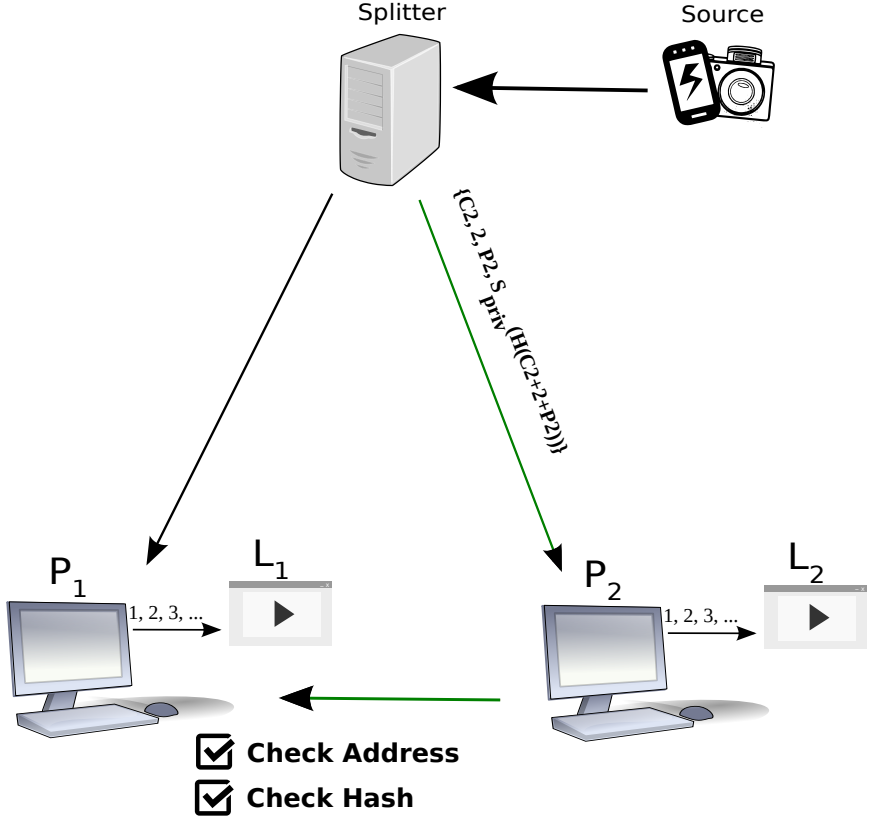
1.WHEN PEERS JOIN THE TEAM THEY RECEIVE THE PUBLIC KEY OF THE SPLITTER.

2.FOR EACH CHUNK, THE SPLITTER SENDS A MESSAGE LIKE THIS:

$$\{chunk, nChunk, dst, Spriv(H(chunk + nChunk + dst))\}$$

STRATEGY BASED ON TRUSTED PEERS AND DIGITAL SIGNATURES

IT HAS BEEN DESIGNED TO MITIGATE THE SELECTIVE ATTACK AND TO IDENTIFY POISONED CHUNKS BY USING DIGITAL SIGNATURES. THE BEHAVIOR RULES ARE:



1.WHEN PEERS JOIN THE TEAM THEY RECEIVE THE PUBLIC KEY OF THE SPLITTER.

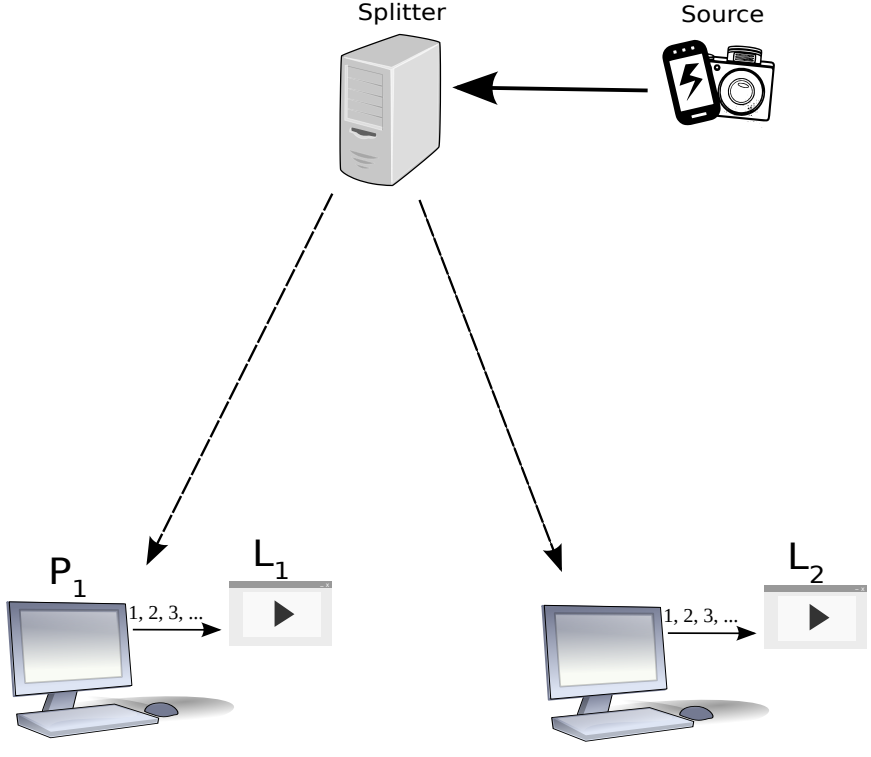
2.FOR EACH CHUNK, THE SPLITTER SENDS A MESSAGE LIKE THIS:

$$\{chunk, nChunk, dst, Spriv(H(chunk + nChunk + dst))\}$$

3.THE PEERS VERIFY DST AND CHECK IF THE HASH VALUE IS CORRECT.

STRATEGY BASED ON TRUSTED PEERS AND DIGITAL SIGNATURES

IT HAS BEEN DESIGNED TO MITIGATE THE SELECTIVE ATTACK AND TO IDENTIFY POISONED CHUNKS BY USING DIGITAL SIGNATURES. THE BEHAVIOR RULES ARE:



1.WHEN PEERS JOIN THE TEAM THEY RECEIVE THE PUBLIC KEY OF THE SPLITTER.

2.FOR EACH CHUNK, THE SPLITTER SENDS A MESSAGE LIKE THIS:

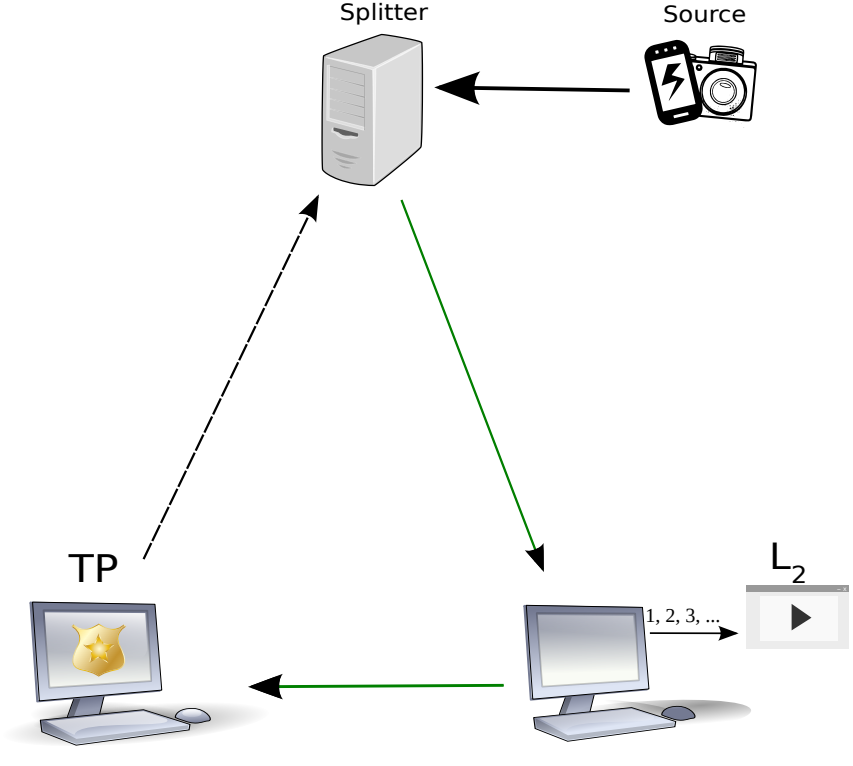
$$\{chunk, nChunk, dst, Spriv(H(chunk + nChunk + dst))\}$$

3.THE PEERS VERIFY DST AND CHECK IF THE HASH VALUE IS CORRECT.

4.THE SPLITTER PERIODICALLY REQUESTS THE LIST OF REMOVED PEERS TO THE TP.

STRATEGY BASED ON TRUSTED PEERS AND DIGITAL SIGNATURES

IT HAS BEEN DESIGNED TO MITIGATE THE SELECTIVE ATTACK AND TO IDENTIFY POISONED CHUNKS BY USING DIGITAL SIGNATURES. THE BEHAVIOR RULES ARE:



1.WHEN PEERS JOIN THE TEAM THEY RECEIVE THE PUBLIC KEY OF THE SPLITTER.

2.FOR EACH CHUNK, THE SPLITTER SENDS A MESSAGE LIKE THIS:

$$\{chunk, nChunk, dst, Spriv(H(chunk + nChunk + dst))\}$$

3.THE PEERS VERIFY DST AND CHECK IF THE HASH VALUE IS CORRECT.

4.THE SPLITTER PERIODICALLY REQUESTS THE LIST OF REMOVED PEERS TO THE TP.

5.PEERS REMOVED BY ANY TP ARE DIRECTLY EXPELLED BY THE SPLITTER AFTER A RANDOM TIME.

STRATEGY BASED ON TRUSTED PEERS AND DIGITAL SIGNATURES

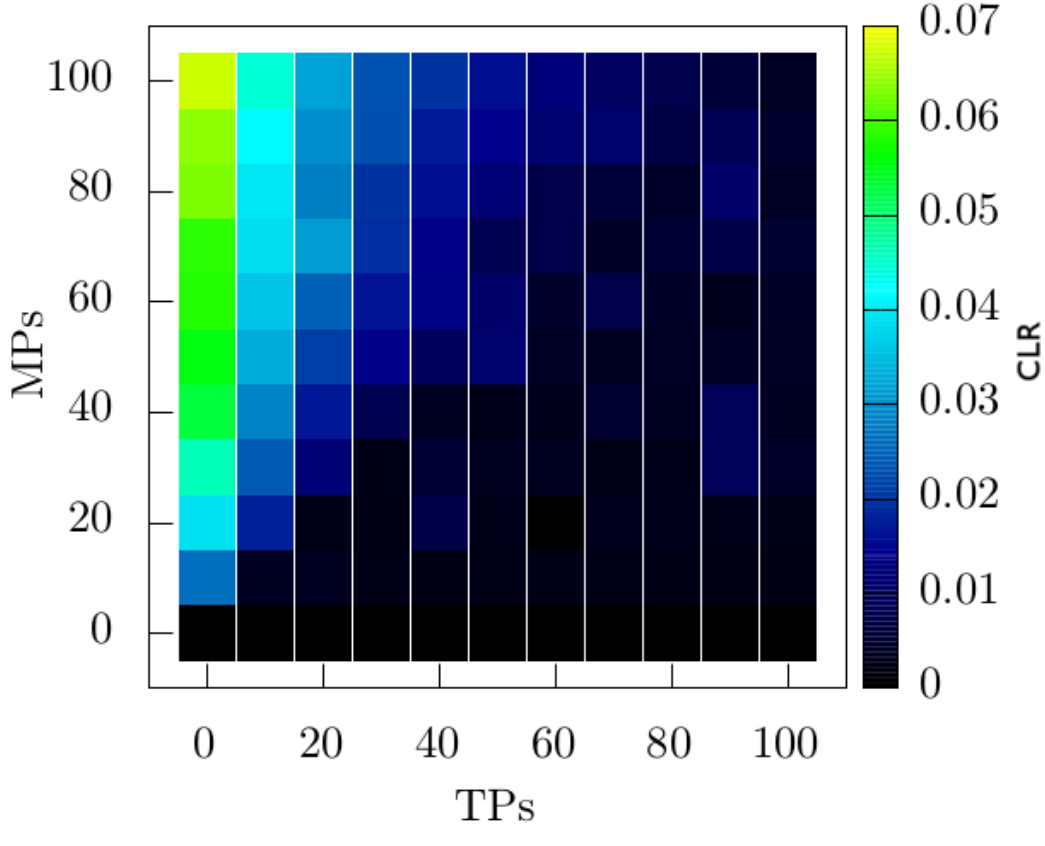
STRATEGY BASED ON TRUSTED PEERS AND DIGITAL SIGNATURES

- **CAN TRUSTED PEERS EXPEL THE BAD GUYS BY FOLLOWING THESE BASIC RULES?**

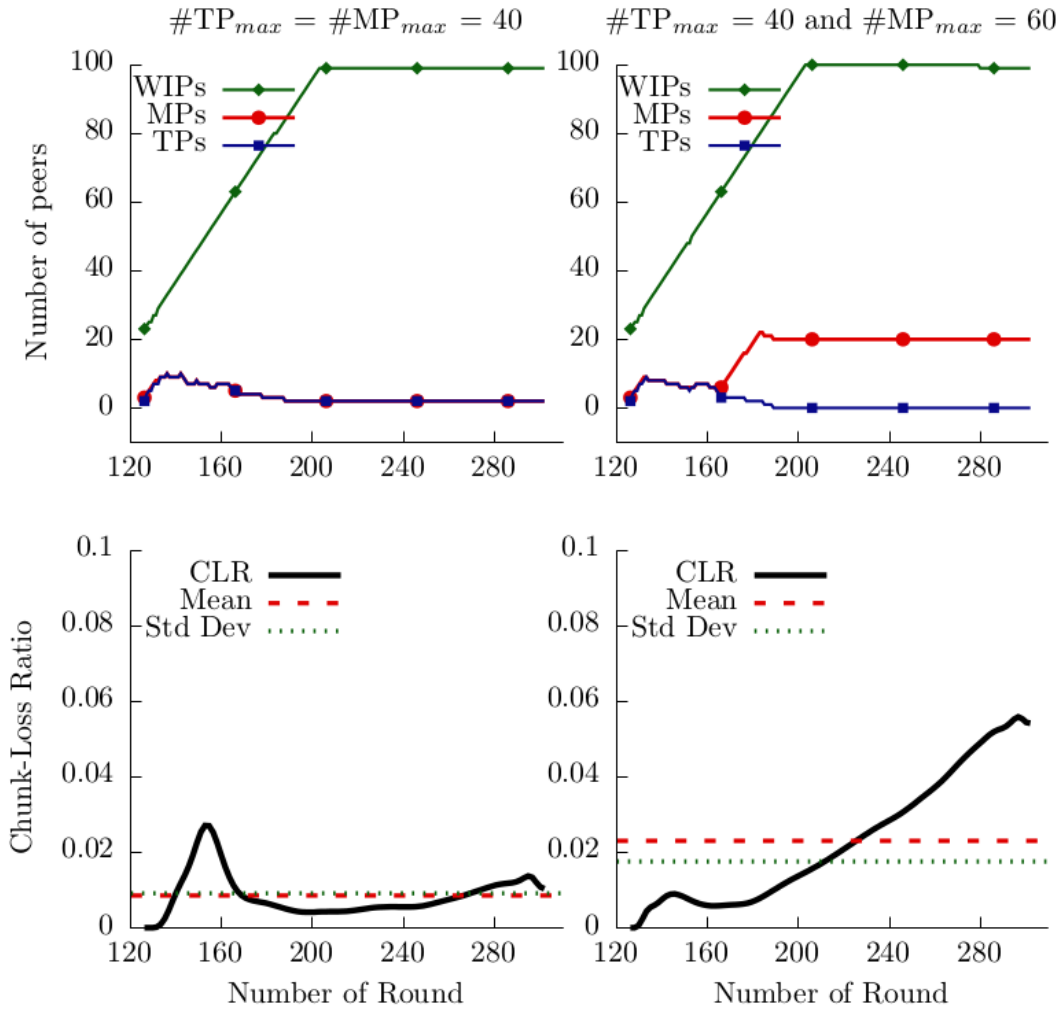
EXPERIMENTAL RESULTS

RESULTS OBTAINED BY SIMULATION [WAR-GAMES]

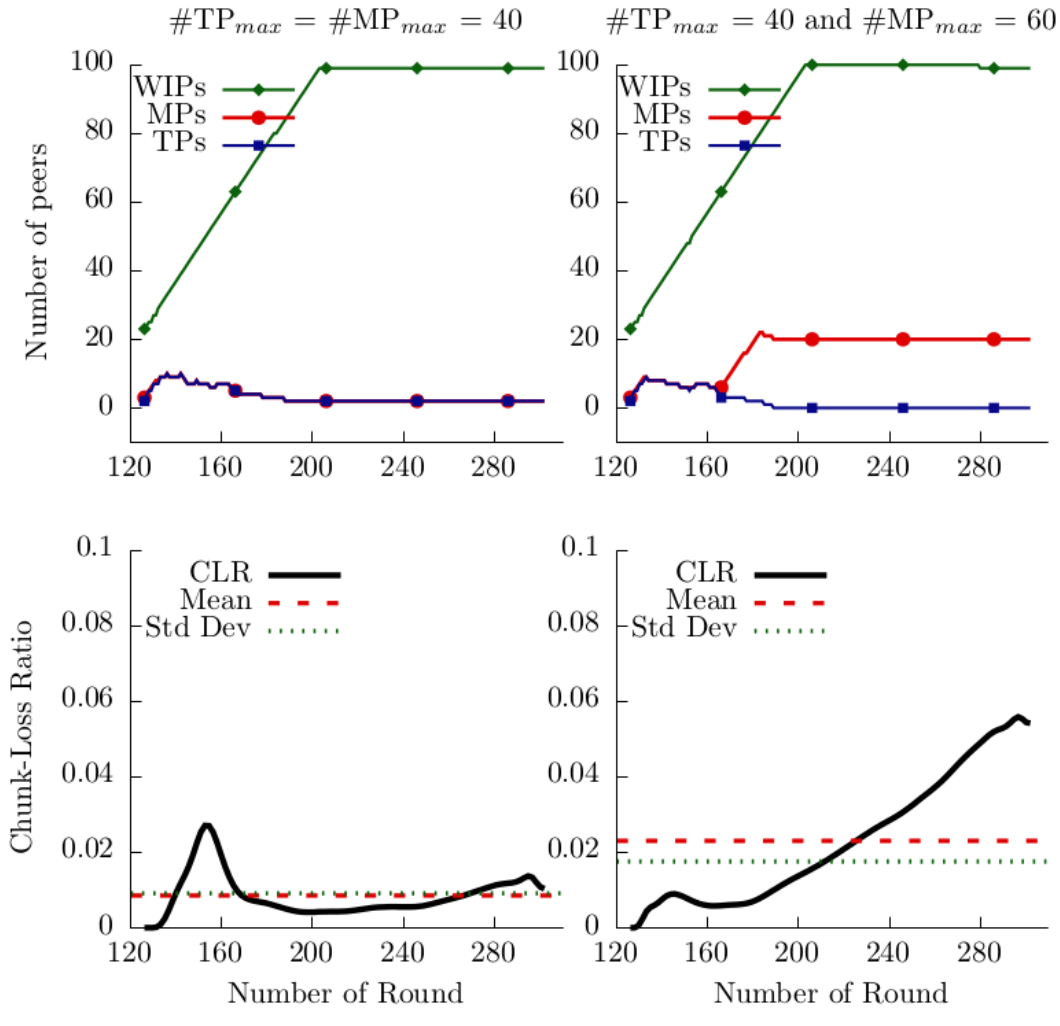
Persistent, Selective and Collaborative Attack



EXPERIMENTAL RESULTS



EXPERIMENTAL RESULTS



Trusted peers as unique defense strategy is not appealing when the number of MPs is large

A HARD MULTI-OBJECTIVE OPTIMIZATION PROBLEM

Attackers and defenders have to solve a multi-objective optimization problem and each decision made by one part may have a countermeasure by the other part

A HARD MULTI-OBJECTIVE OPTIMIZATION PROBLEM

Attackers and defenders have to solve a multi-objective optimization problem and each decision made by one part may have a countermeasure by the other part

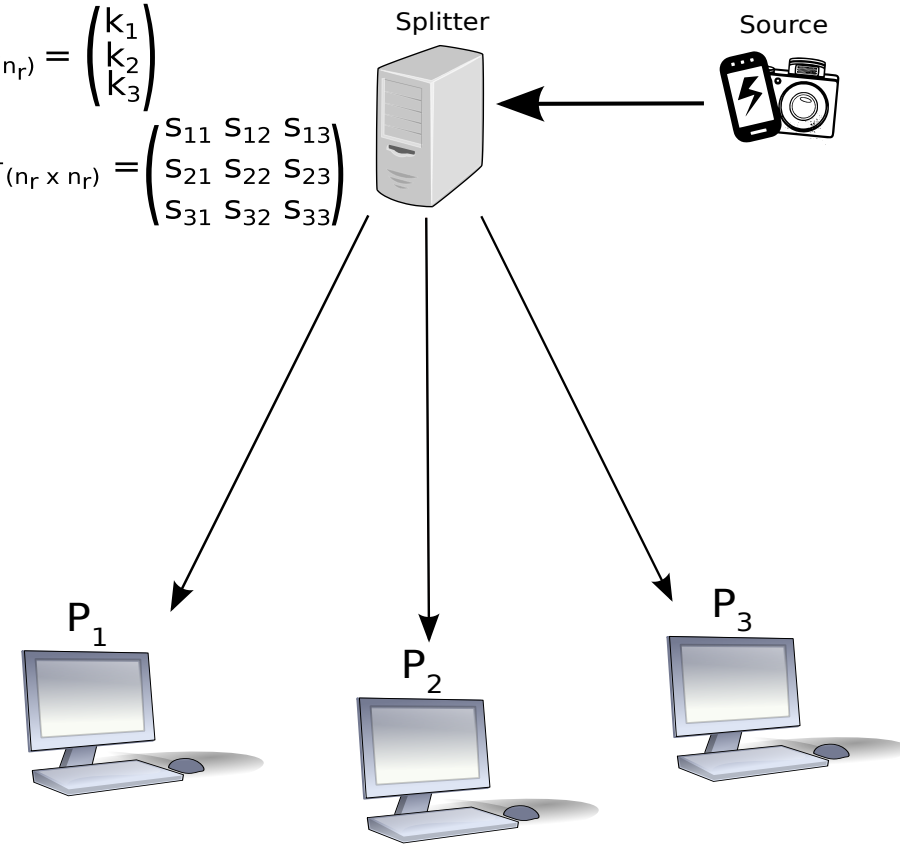
WHO ARE THE GOOD/BAD GUYS?

STRATEGY BASED ON TRUSTED PEERS AND SHAMIR'S SECRET SHARING

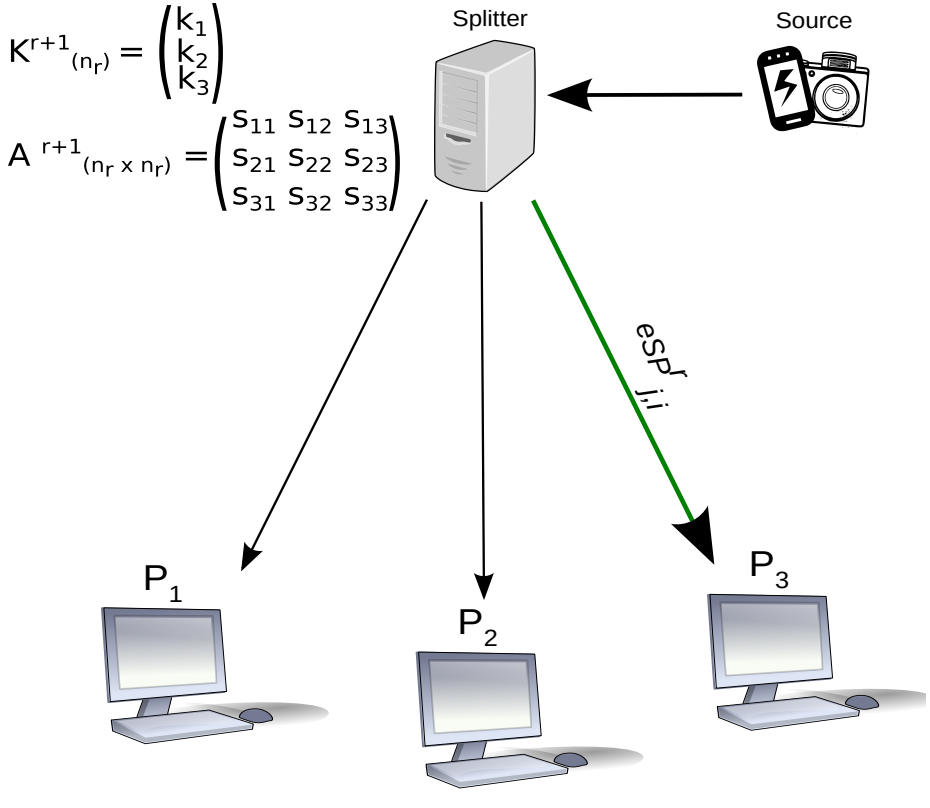
THE MAIN IDEA IS VERY SIMPLE: “IF YOU WANT TO REMAIN IN THE TEAM YOU MUST HAVE A GOOD BEHAVIOR WITH AT LEAST T PEERS”. THE BEHAVIOR RULES ARE:

$$K^{r+1}_{(n_r)} = \begin{pmatrix} k_1 \\ k_2 \\ k_3 \end{pmatrix}$$

$$A^{r+1}_{(n_r \times n_r)} = \begin{pmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{pmatrix}$$



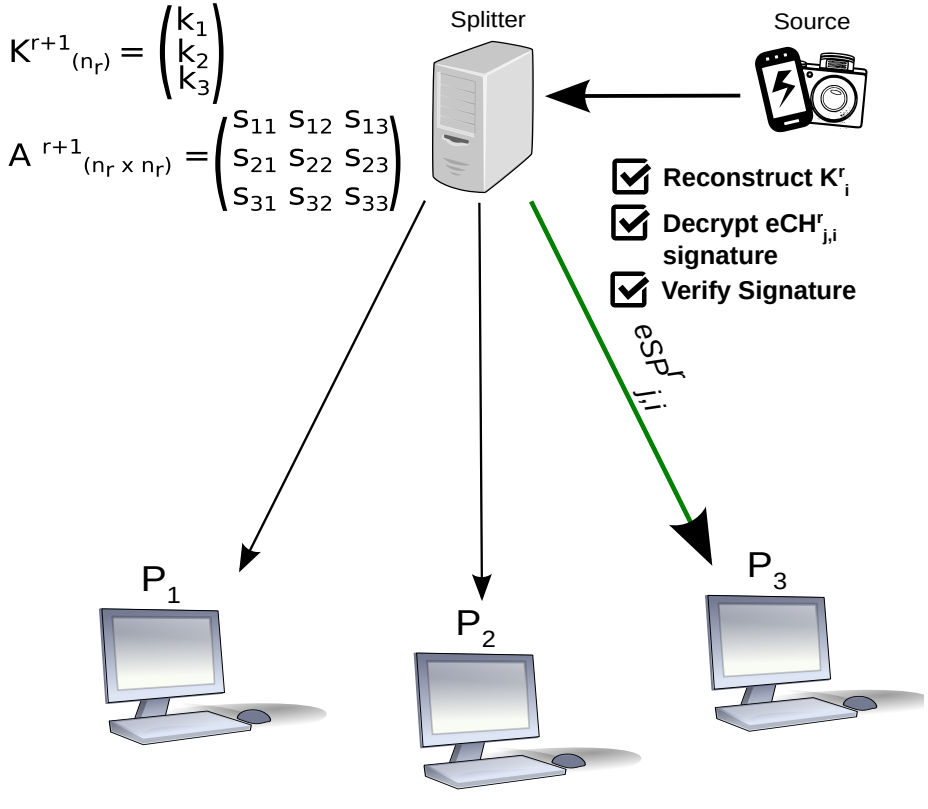
STRATEGY BASED ON TRUSTED PEERS AND SHAMIR'S SECRET SHARING



1.THE SPLITTER SENDS A MESSAGE $eSP^r_{j,i}$.

$$eSP^r_{j,i} = \{eCH^r_{j,i},\ SH^{r+1}_i\}$$
$$eCH^r_{j,i} = \{C_j, j, P_i, r, E_{K^r_i}[S_{pr}(H(C_j||j||P_i||r))]\}$$
$$SH^{r+1}_i = \{\{SH^{r+1}_i\}_q,\ q = 1,\ \ldots,\ n_r\}$$
$$\{SH^{r+1}_i\}_q = \{P_i, P_q, r+1, A^{r+1}_{q,i}, S_{pr}(H(P_i||P_q||r+1||A^{r+1}_{q,i}))\}$$

STRATEGY BASED ON TRUSTED PEERS AND SHAMIR'S SECRET SHARING

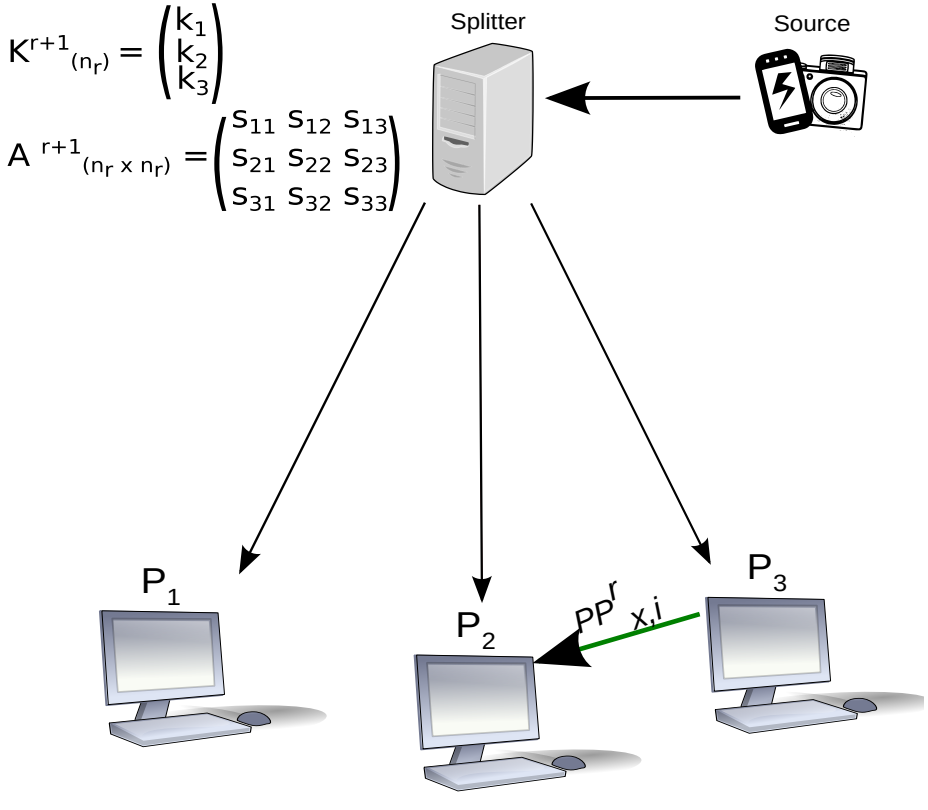


1.THE SPLITTER SENDS A MESSAGE $eSP^r_{j,i}$.

$$eSP^r_{j,i} = \{eCH^r_{j,i},\ SH^{r+1}_i\}$$
$$eCH^r_{j,i} = \{C_j, j, P_i, r, E_{K^r_i}[S_{pr}(H(C_j||j||P_i||r))]\}$$
$$SH^{r+1}_i = \{\{SH^{r+1}_i\}_q,\ q = 1,\ \ldots,\ n_r\}$$
$$\{SH^{r+1}_i\}_q = \{P_i, P_q, r+1, A^{r+1}_{q,i}, S_{pr}(H(P_i||P_q||r+1||A^{r+1}_{q,i}))\}$$

2.THE PEER RECONSTRUCTS K^r_i, DECRYTPS THE MESSAGE AND VERIFIES THE SIGNATURE.

STRATEGY BASED ON TRUSTED PEERS AND SHAMIR'S SECRET SHARING



1.THE SPLITTER SENDS A MESSAGE $eSP^r_{j,i}$.

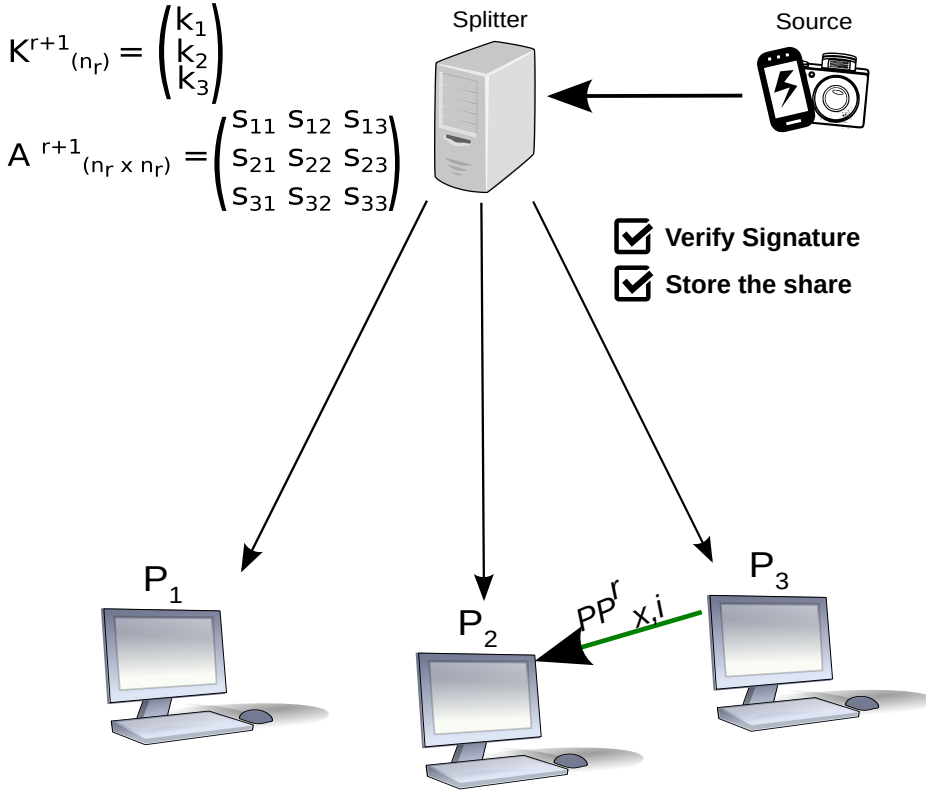
$$eSP^r_{j,i} = \{eCH^r_{j,i},\ SH^{r+1}_i\}$$
$$eCH^r_{j,i} = \{C_j, j, P_i, r, E_{K^r_i}[S_{pr}(H(C_j||j||P_i||r))]\}$$
$$SH^{r+1}_i = \{\{SH^{r+1}_i\}_q,\ q = 1,\ \ldots,\ n_r\}$$
$$\{SH^{r+1}_i\}_q = \{P_i, P_q, r+1, A^{r+1}_{q,i}, S_{pr}(H(P_i||P_q||r+1||A^{r+1}_{q,i}))\}$$

2.THE PEER RECONSTRUCTS K^r_i, DECRYTPS THE MESSAGE AND VERIFIES THE SIGNATURE.

3.THE PEER SENDS THE MESSAGE DECRYPTED.

$$PP^r_{i,q} = \left\{CH^r_{j,i},\ \{SH^{r+1}_i\}_q\right\}$$
$$CH^r_{j,i} = \{C_j, j, P_i, r, S_{pr}(H(C_j||j||P_i||r))\}$$

STRATEGY BASED ON TRUSTED PEERS AND SHAMIR'S SECRET SHARING



1.THE SPLITTER SENDS A MESSAGE $eSP^r_{j,i}$.

$$eSP^r_{j,i} = \{eCH^r_{j,i},\ SH^{r+1}_i\}$$
$$eCH^r_{j,i} = \{C_j, j, P_i, r, E_{K^r_i}[S_{pr}(H(C_j||j||P_i||r))]\}$$
$$SH^{r+1}_i = \{\{SH^{r+1}_i\}_q,\ q = 1,\ \ldots,\ n_r\}$$
$$\{SH^{r+1}_i\}_q = \{P_i, P_q, r+1, A^{r+1}_{q,i}, S_{pr}(H(P_i||P_q||r+1||A^{r+1}_{q,i}))\}$$

2.THE PEER RECONSTRUCTS K^r_i, DECRYTPS THE MESSAGE AND VERIFIES THE SIGNATURE.

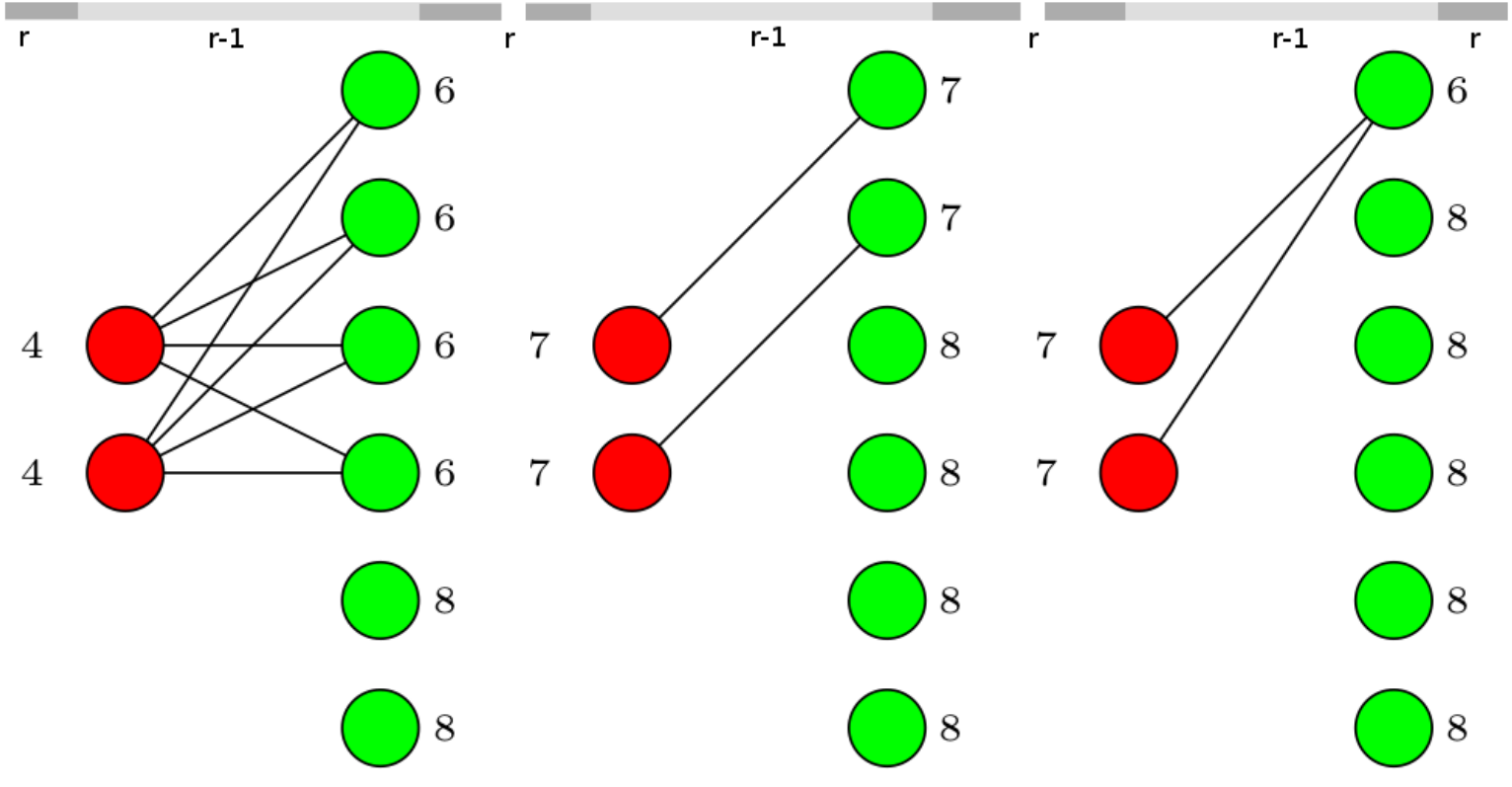
3.THE PEER SENDS THE MESSAGE DECRYPTED.

$$PP^r_{i,q} = \left\{CH^r_{j,i},\ \{SH^{r+1}_i\}_q\right\}$$
$$CH^r_{j,i} = \{C_j, j, P_i, r, S_{pr}(H(C_j||j||P_i||r))\}$$

4.THE PEER VERIFIES THE MESSAGE AND SAVES THE SHARE.

THEORETICAL RESULTS

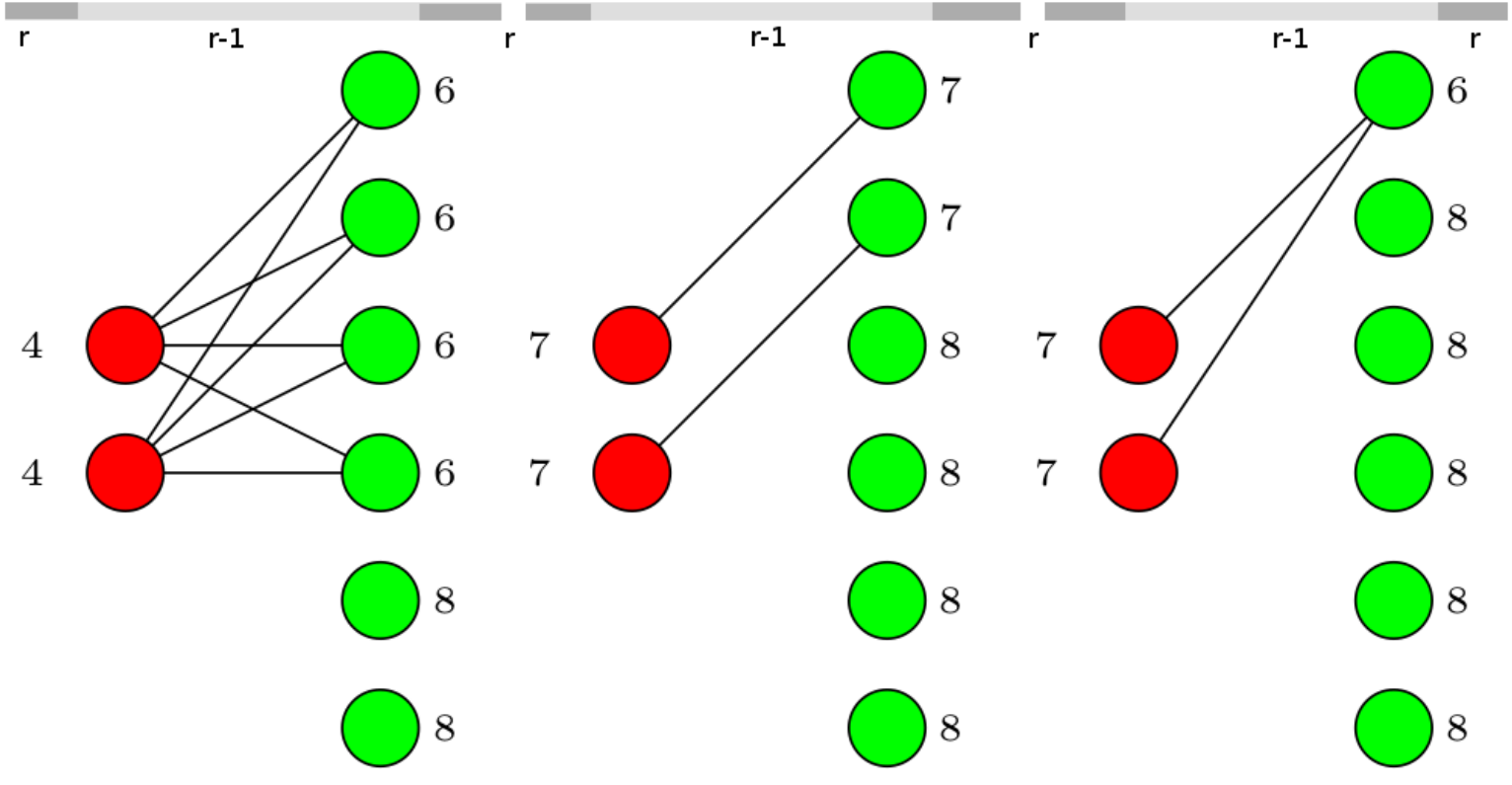
RESULTS OBTAINED AFTER A THEORETICAL ANALYSIS: $\mathrm{MPs} <= N/2$



(a) One MP to many (b) One MP to one (c) Many MPs to one

THEORETICAL RESULTS

RESULTS OBTAINED AFTER A THEORETICAL ANALYSIS: $\mathrm{MPs} <= N/2$



(a) One MP to many (b) One MP to one (c) Many MPs to one

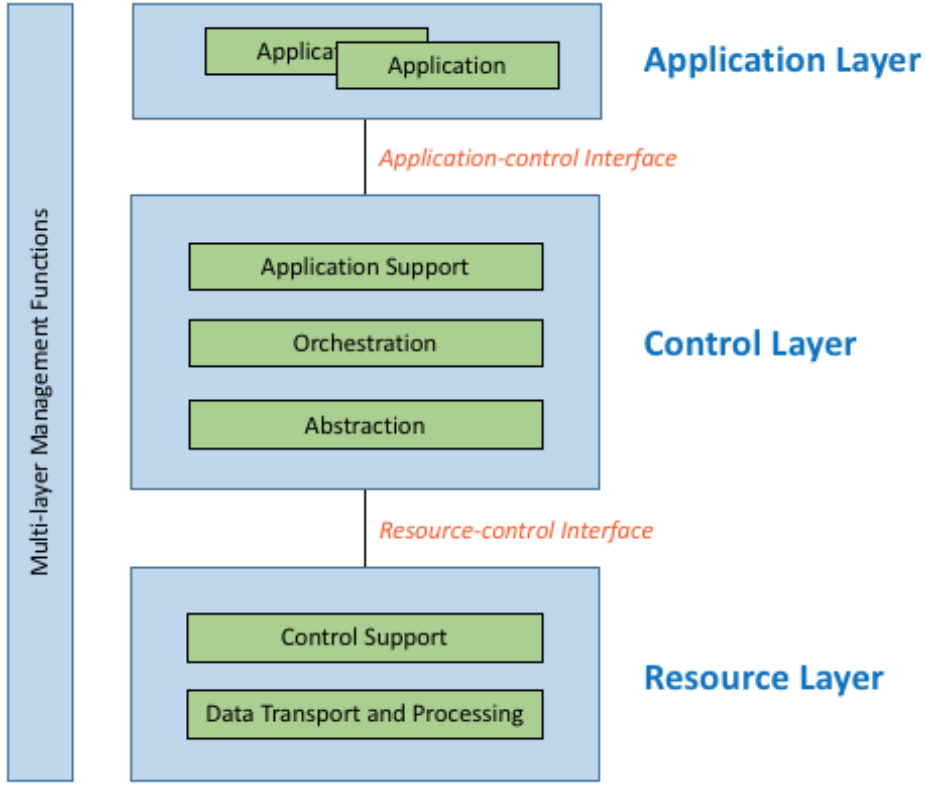
Most severe attack is fully mitigated with only one Trusted Peer

OUTLINE

- INTRODUCTION
  - Background
  - Research Questions
- PEER TO PEER STRAIGHFORWARD PROTOCOL
  - An application-layer protocol that provides real-time broadcasting on the Internet
- **SECURITY**
  - Pollution Attacks
  - Strategies applied to Traditional Networks
  - Strategies applied to Software Defined Networks
- EFFICIENCY
  - NAT Traversal using Collaborative Port Prediction
  - Runing P2PSP on Embeded Devices
- CONCLUSIONS
  - Key Results and Future Work

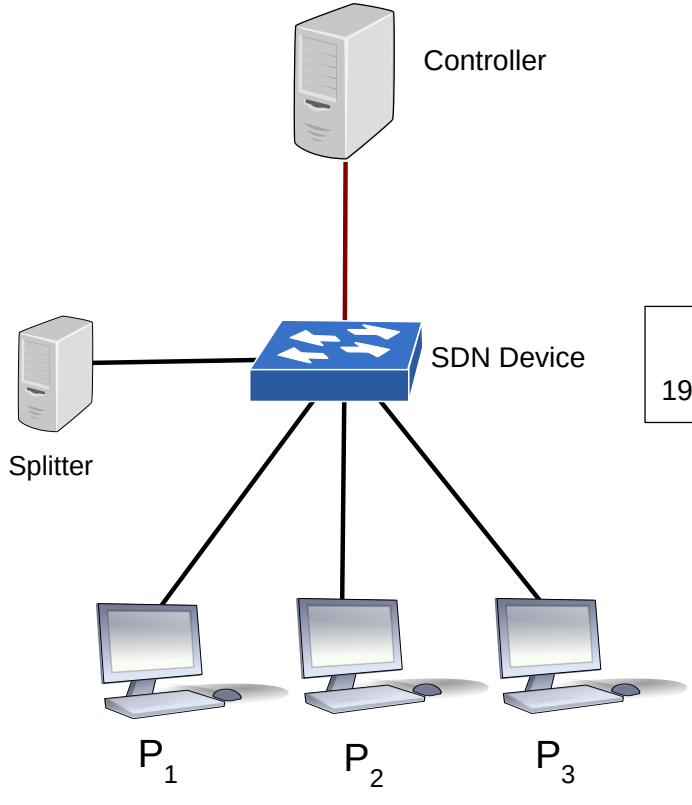
SOFTWARE DEFINED NETWORKS

SDN ARCHITECTURE



SOFTWARE DEFINED NETWORKS

P2PSP PROPOSAL



Ryu Controller
Scrambling table:
(.1,.1), (.2,.3), (.3,.2)

Splitter
192.168.0.99

OF Switch

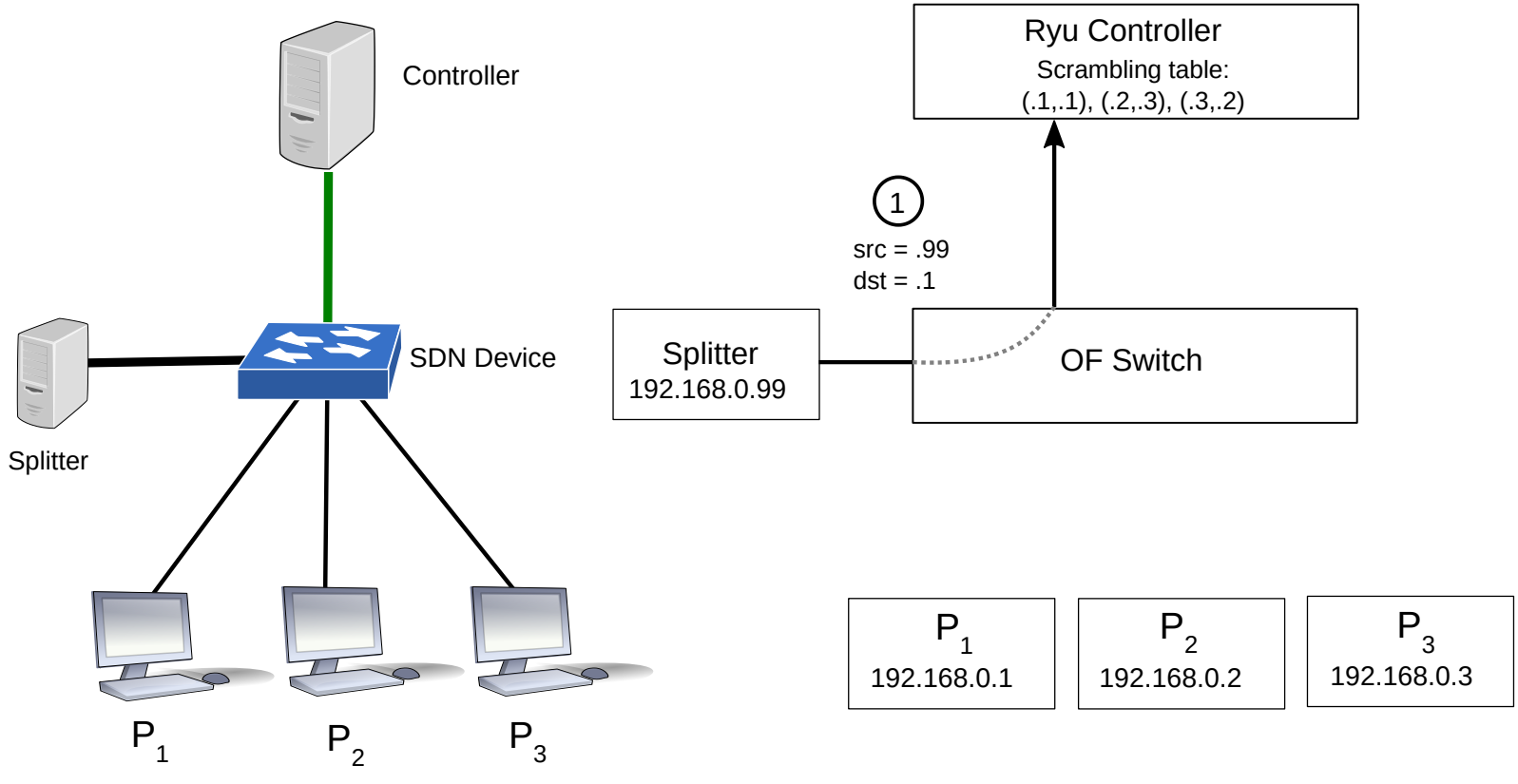
P_1
192.168.0.1

P_2
192.168.0.2

P_3
192.168.0.3

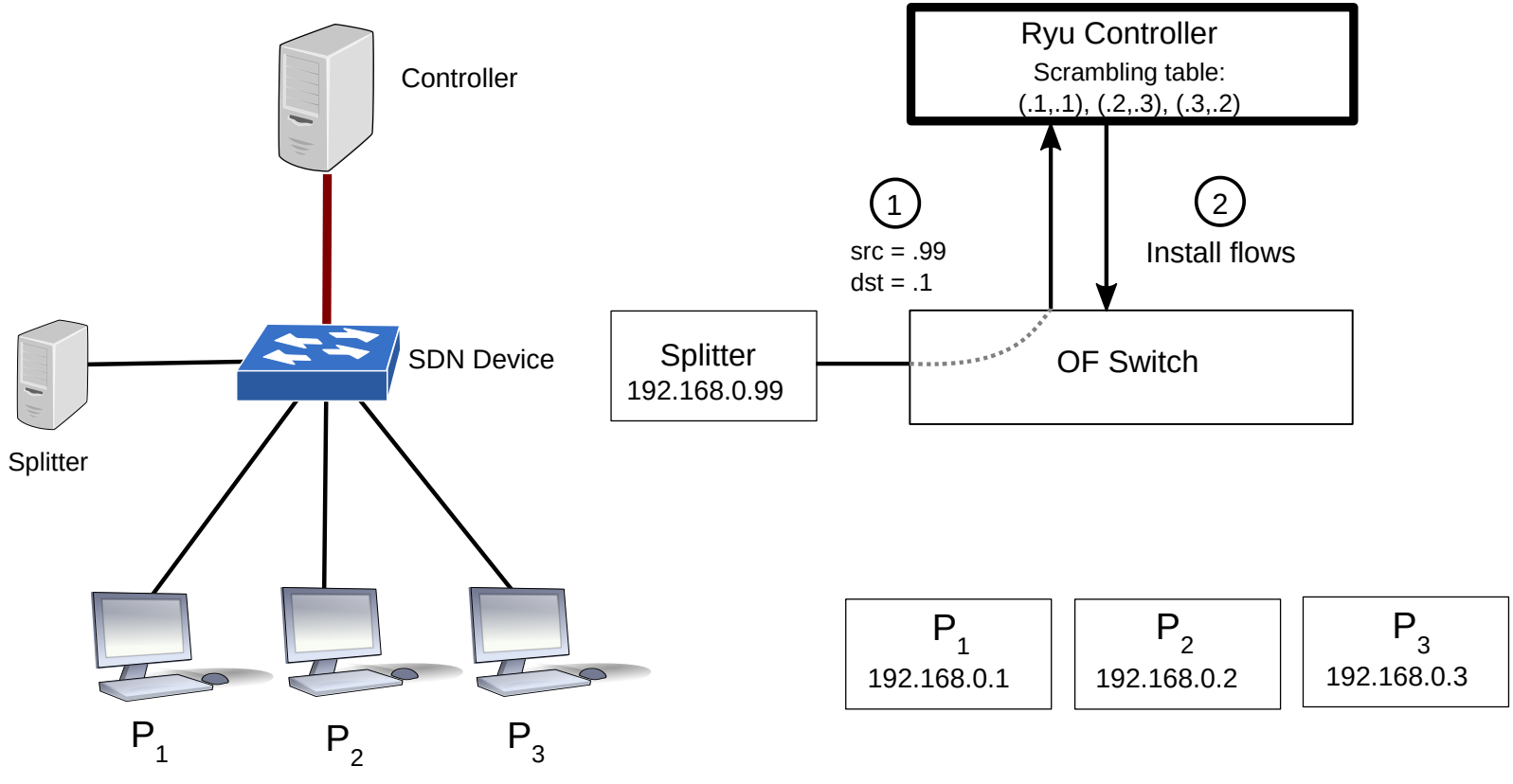
SOFTWARE DEFINED NETWORKS

P2PSP PROPOSAL



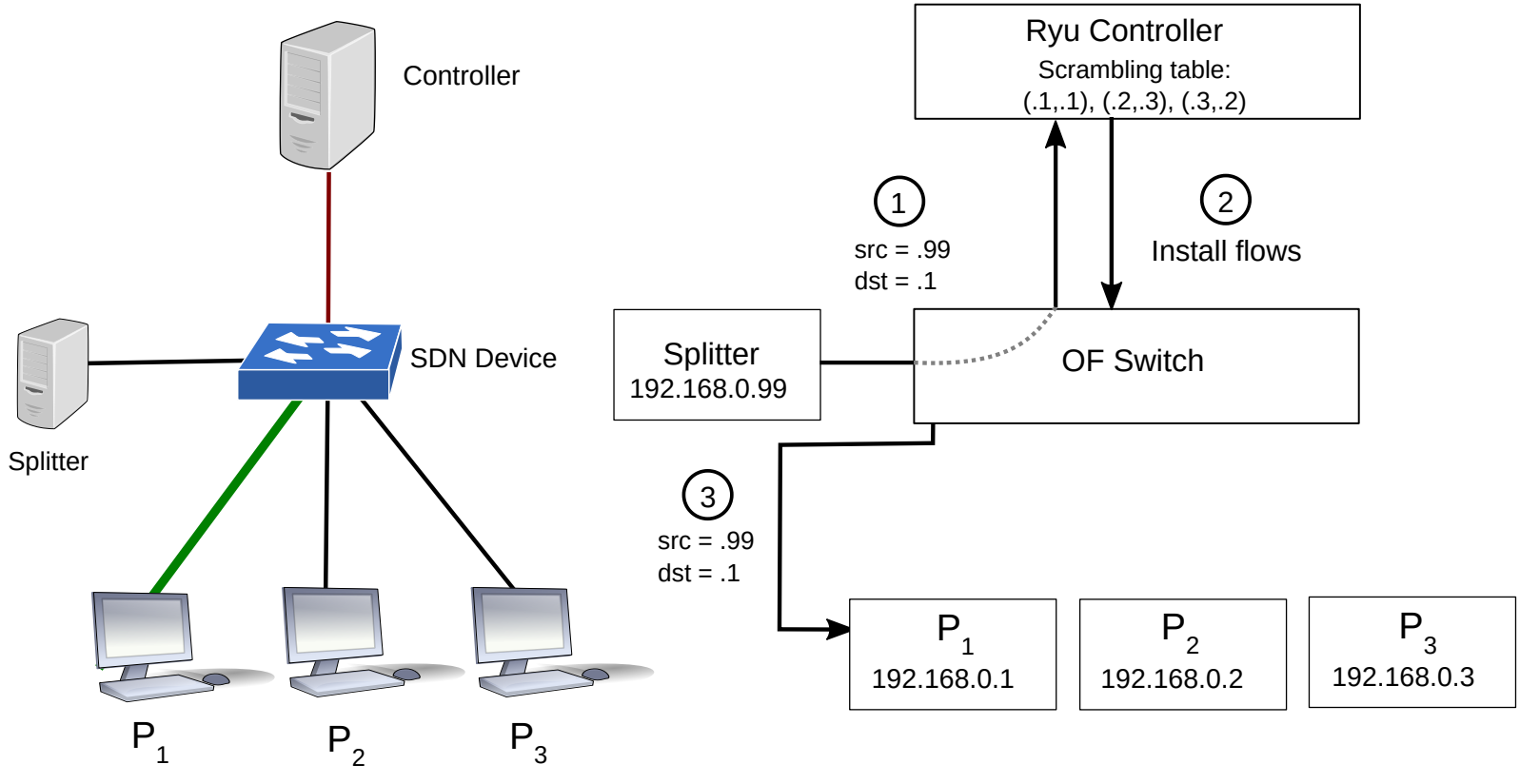
SOFTWARE DEFINED NETWORKS

P2PSP PROPOSAL



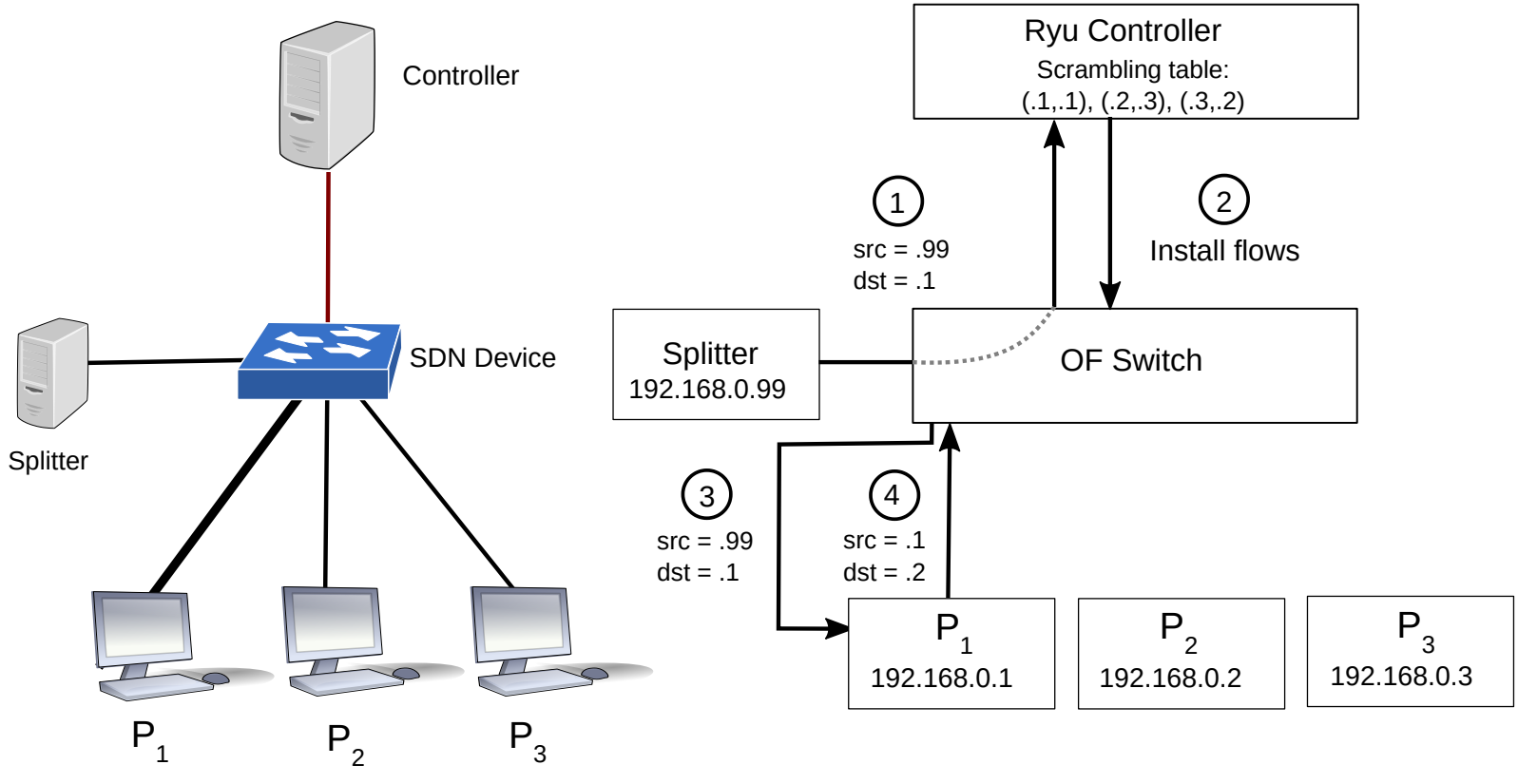
SOFTWARE DEFINED NETWORKS

P2PSP PROPOSAL



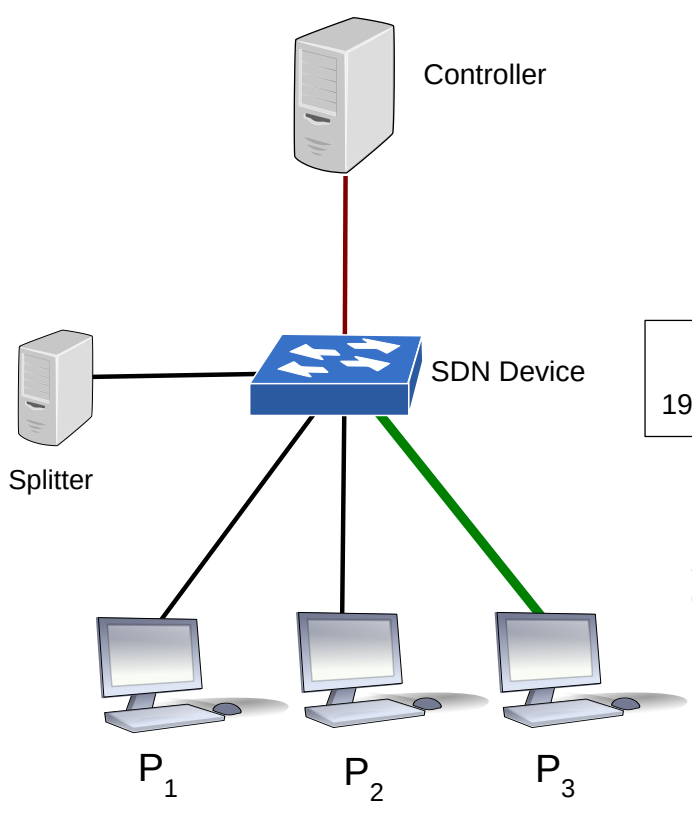
SOFTWARE DEFINED NETWORKS

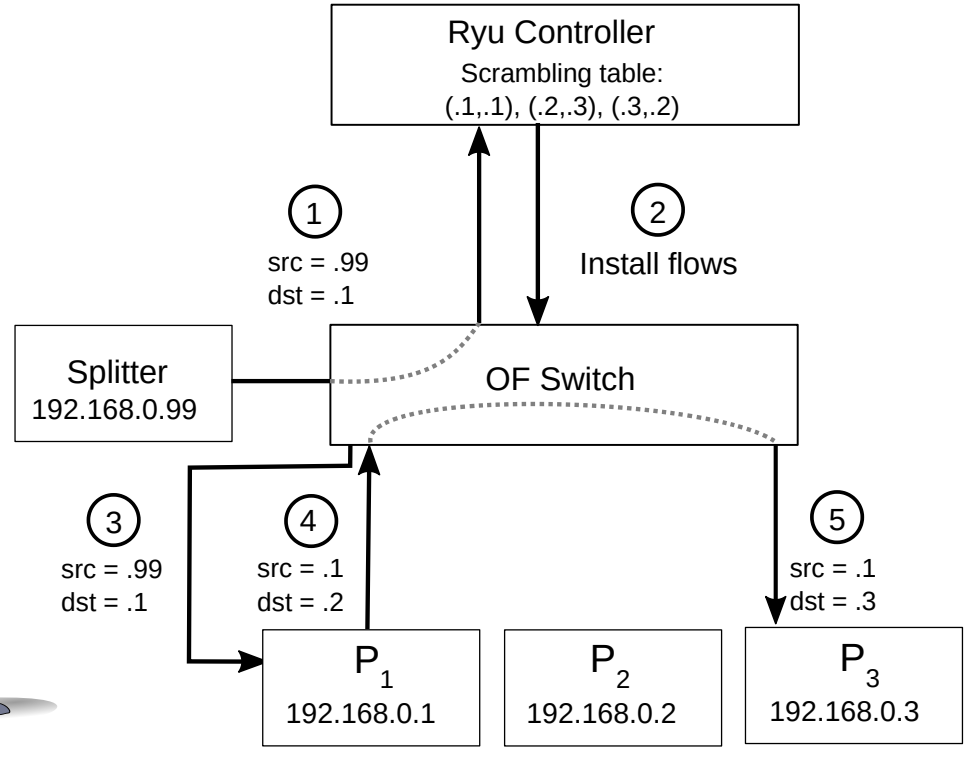
P2PSP PROPOSAL



SOFTWARE DEFINED NETWORKS

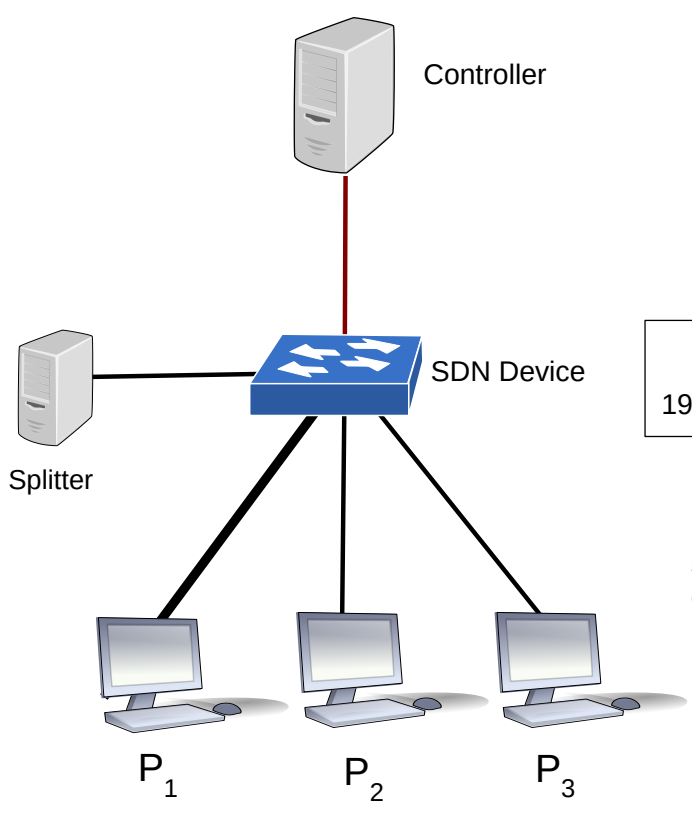
P2PSP PROPOSAL

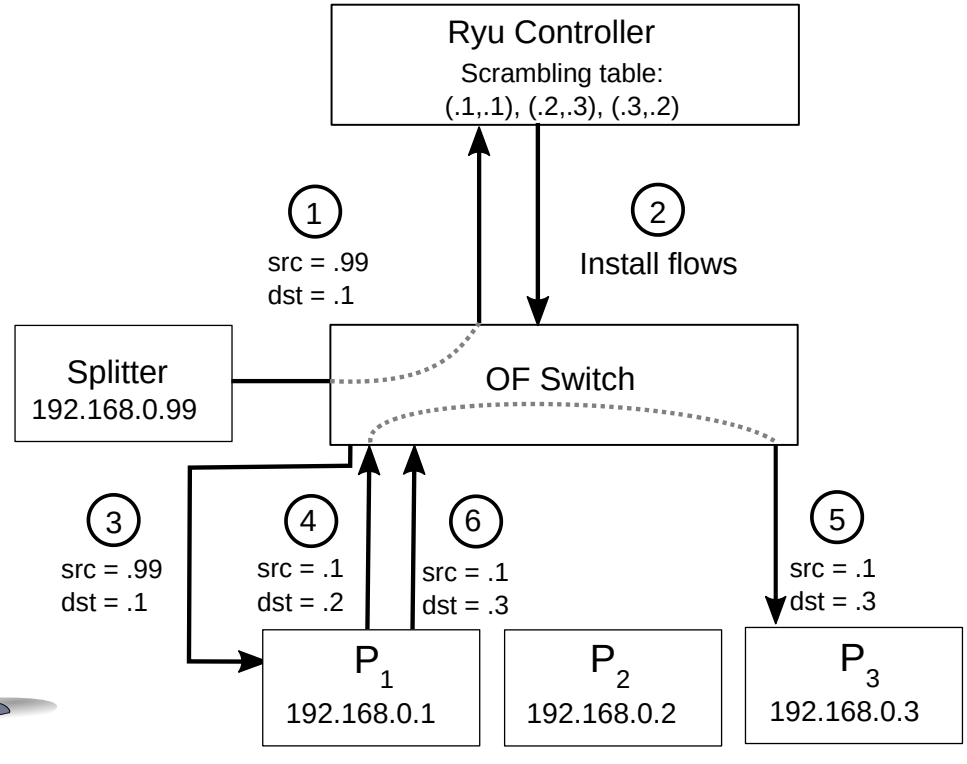




SOFTWARE DEFINED NETWORKS

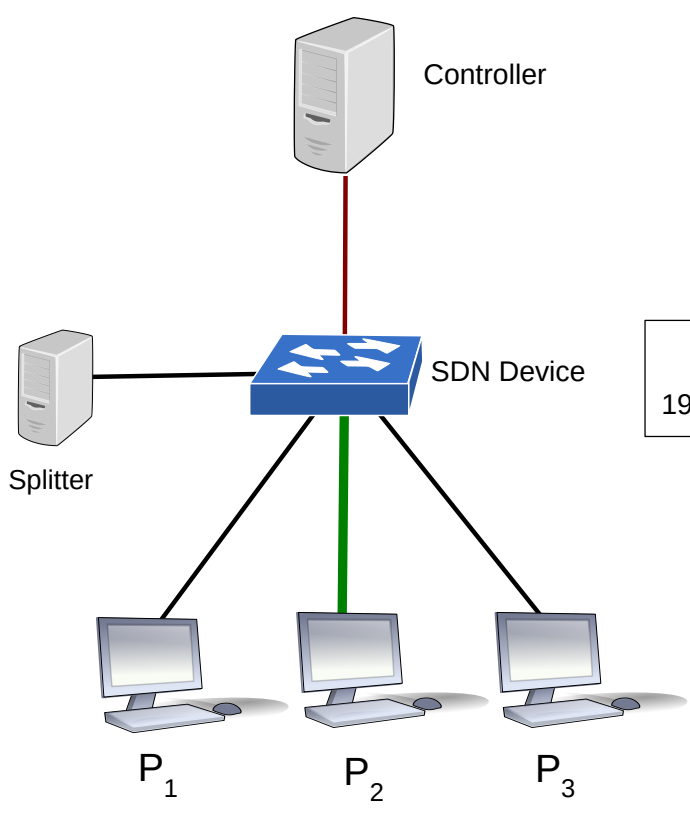
P2PSP PROPOSAL

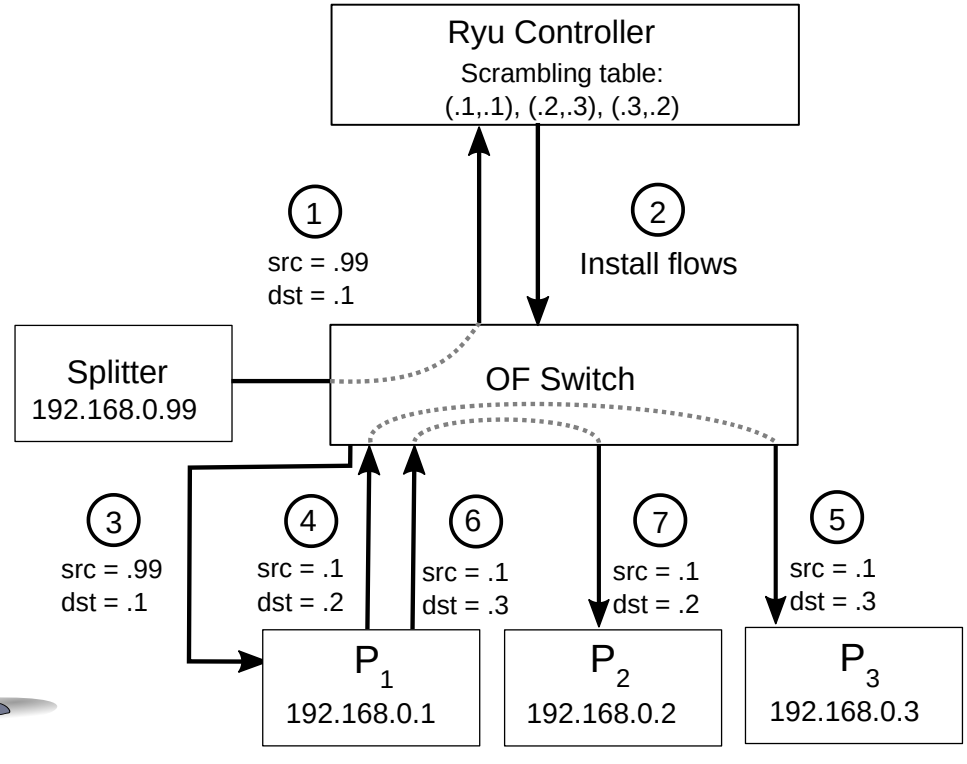




SOFTWARE DEFINED NETWORKS

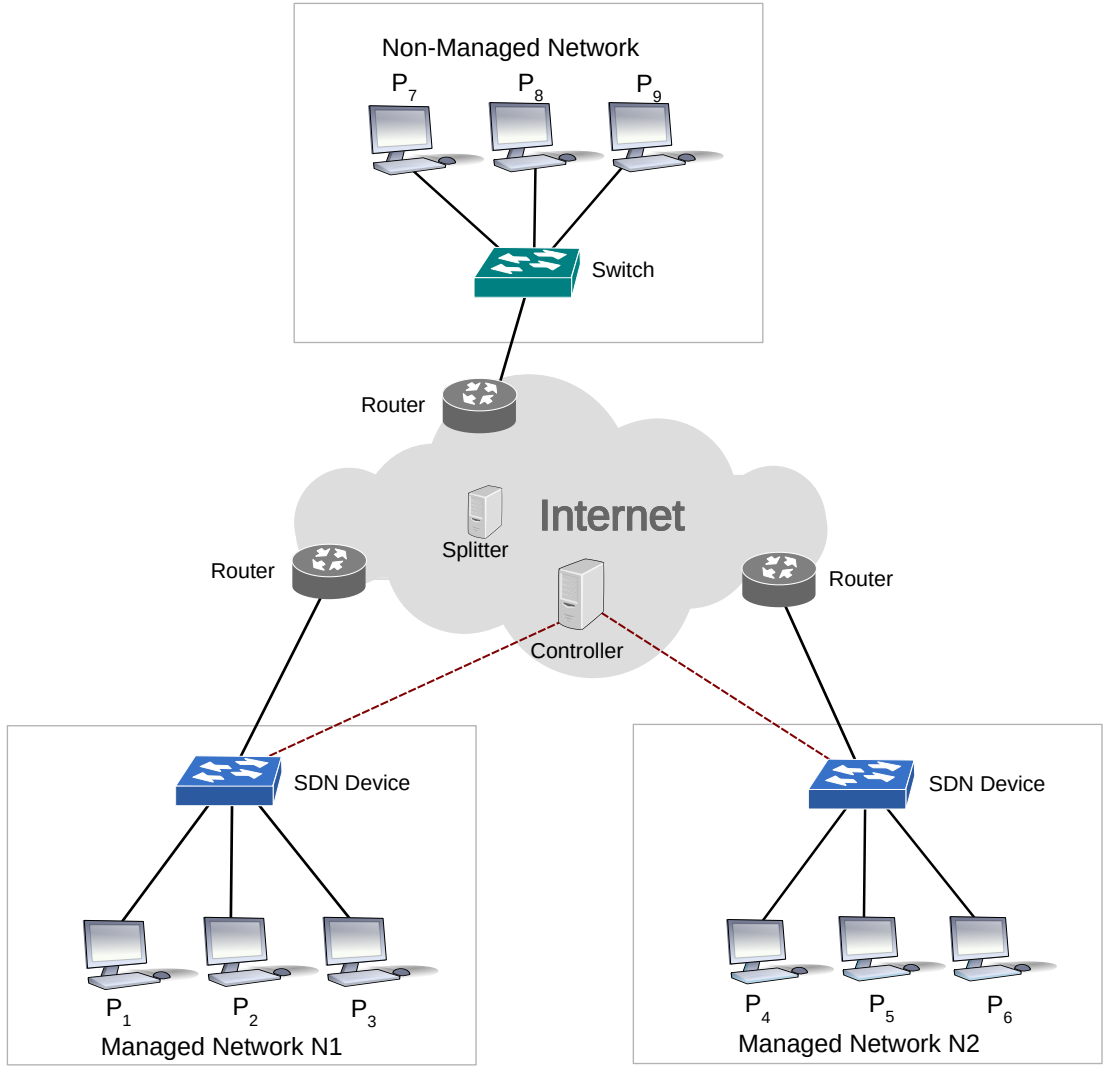
P2PSP PROPOSAL





SOFTWARE DEFINED NETWORKS

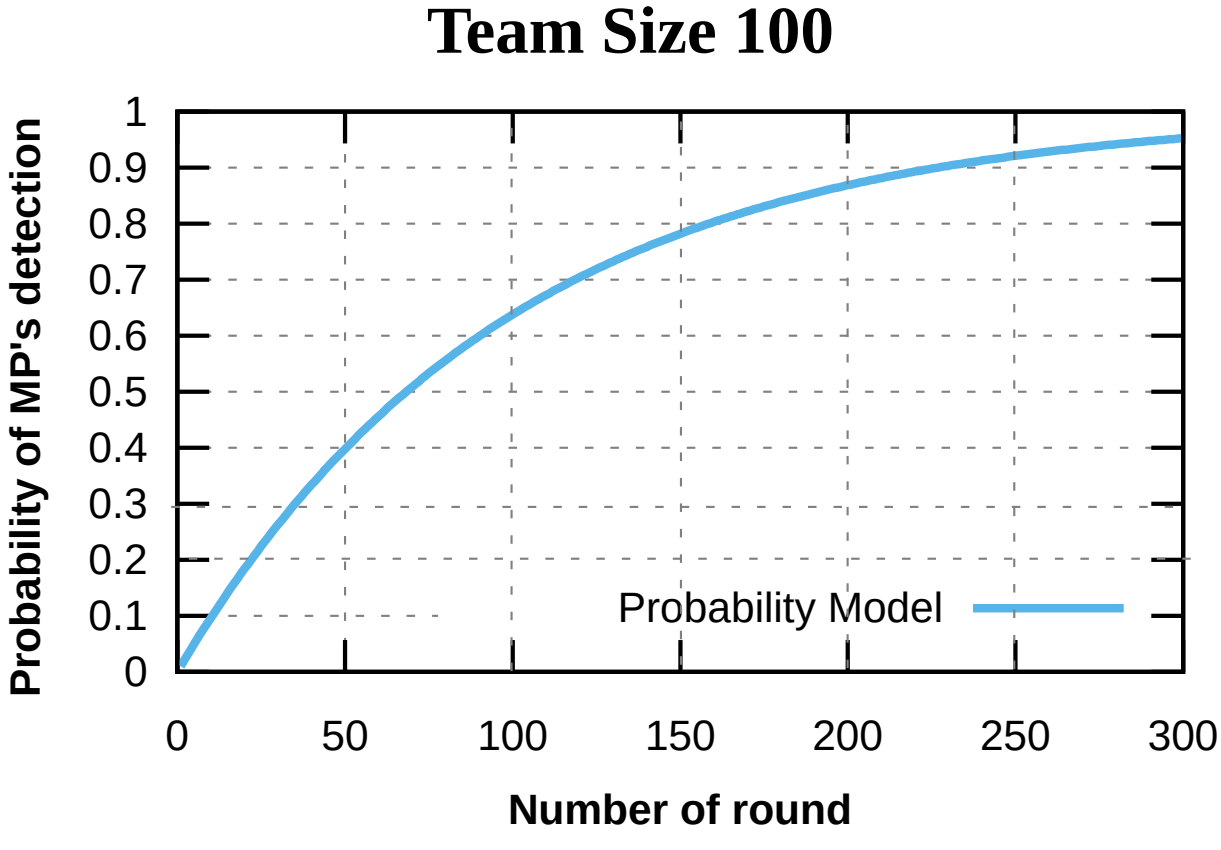
It also works under hybrid environments



DETECTION PROBABILITIES

MODEL

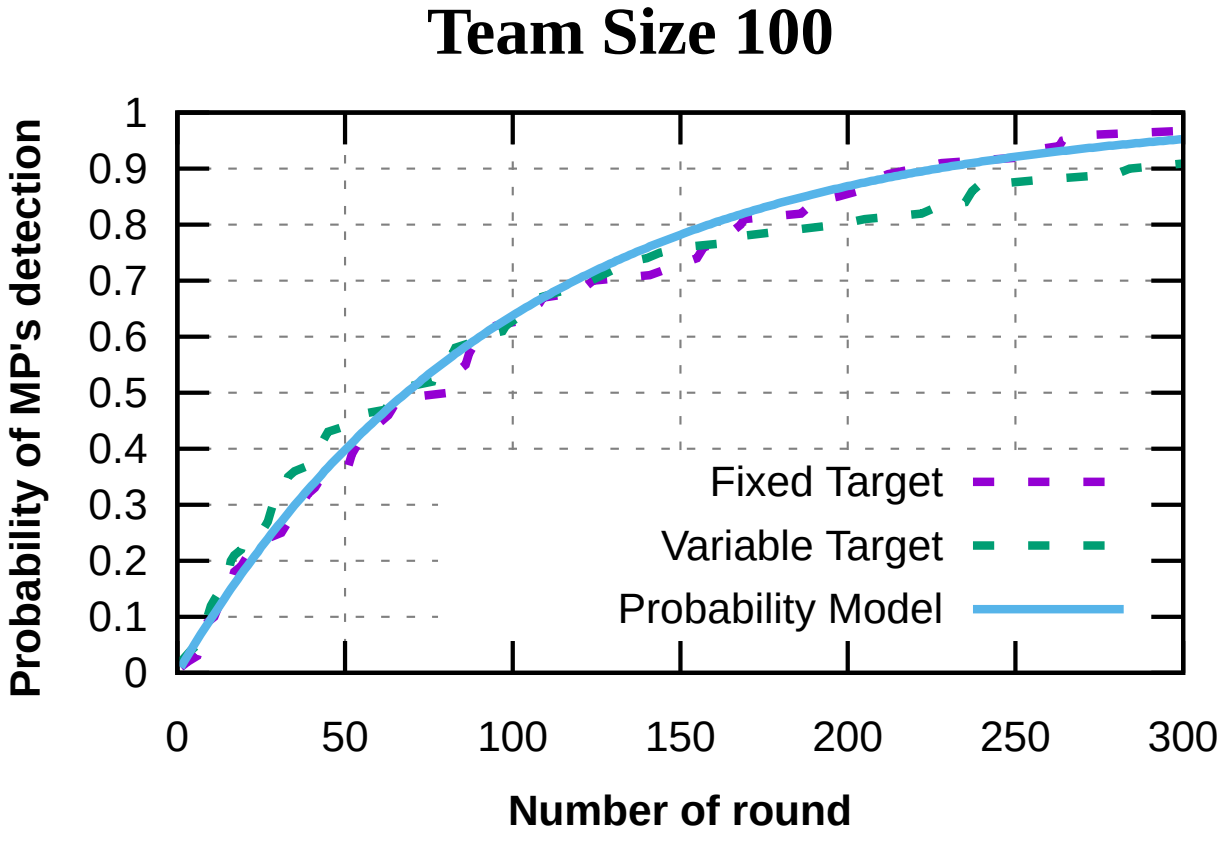
$$p = \sum_{r=1}^{N} \left(\left(\frac{Z-2}{Z-1} \right)^{(r-1)} \times \frac{1}{Z-1} \right)$$



EXPERIMENTS

Team size: 10 to 100 | **Samples:** 100 | **Period:** 1 | **Cases:** Fixed and Variable attacks.

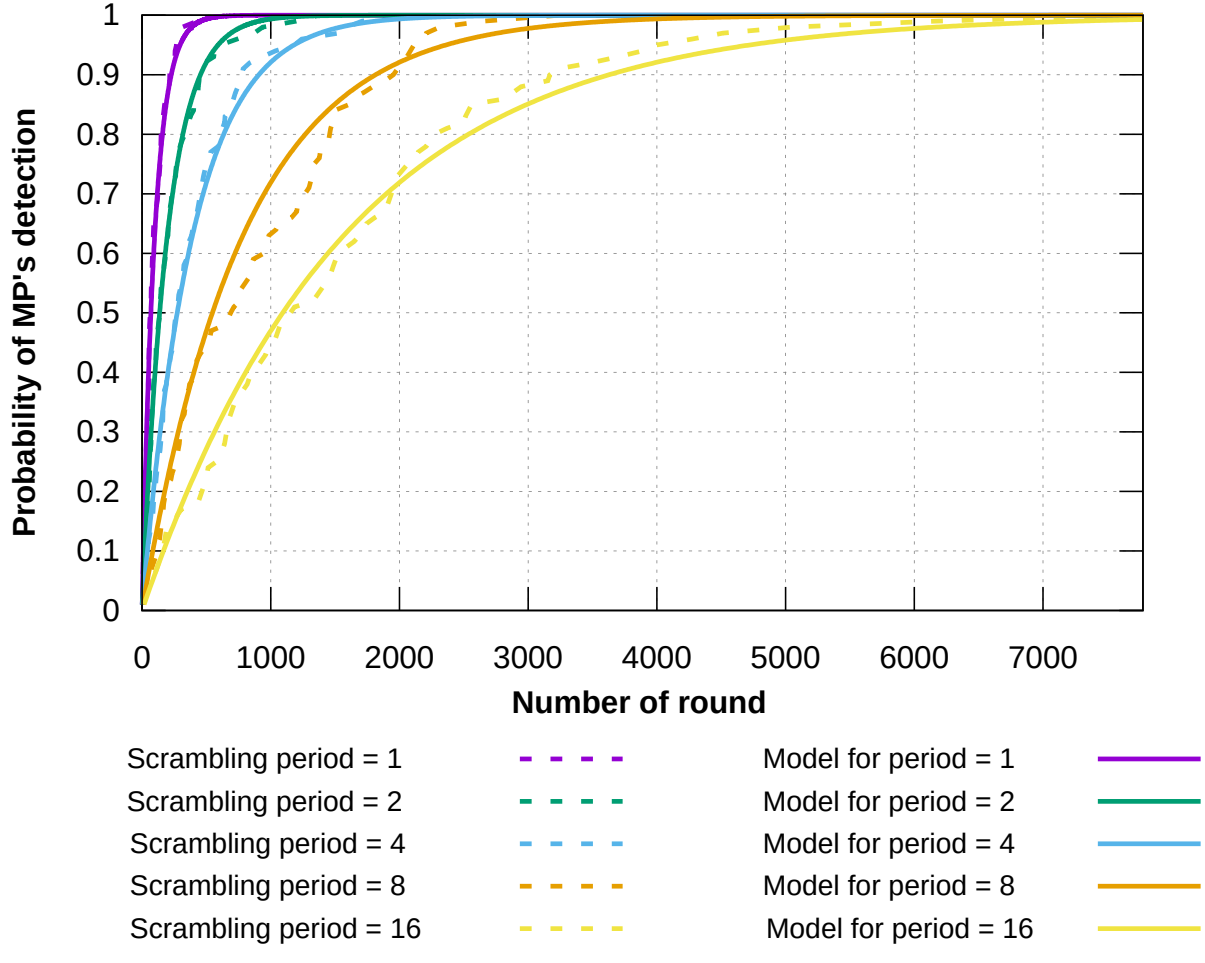
An open-source implementation is available on GitHub



EXPERIMENTS

Modifying the scrambling period

Team Size 100



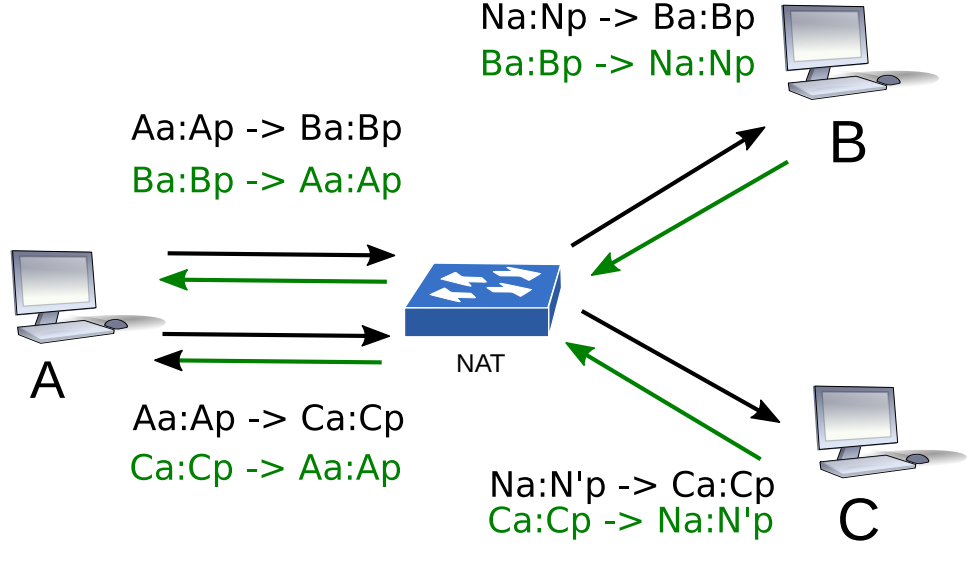
OUTLINE

- INTRODUCTION
  - Background
  - Research Questions
- PEER TO PEER STRAIGHFORWARD PROTOCOL
  - An application-layer protocol that provides real-time broadcasting on the Internet
- SECURITY
  - Pollution Attacks
  - Strategies applied to Traditional Networks
  - Strategies applied to Software Defined Networks
- **EFFICIENCY**
  - NAT Traversal using Collaborative Port Prediction
  - Runing P2PSP on Embeded Devices
- CONCLUSIONS
  - Key Results and Future Work

NAT TRAVERSAL

NAT

It is a method of remapping one IP address space into another by modifying network address information



SYMMETRIC NAT

Each request from the same internal endpoint to a destination endpoint is mapped to a unique external endpoint; if the destination change, a different mapping is used.

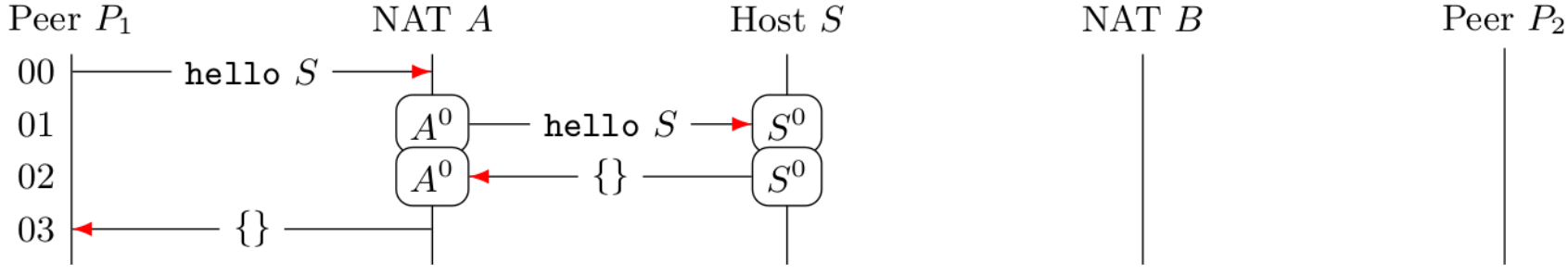
NAT TRAVERSAL

PEERS COMMUNICATION ISSUES

Peer P_1 NAT A Host S NAT B Peer P_2

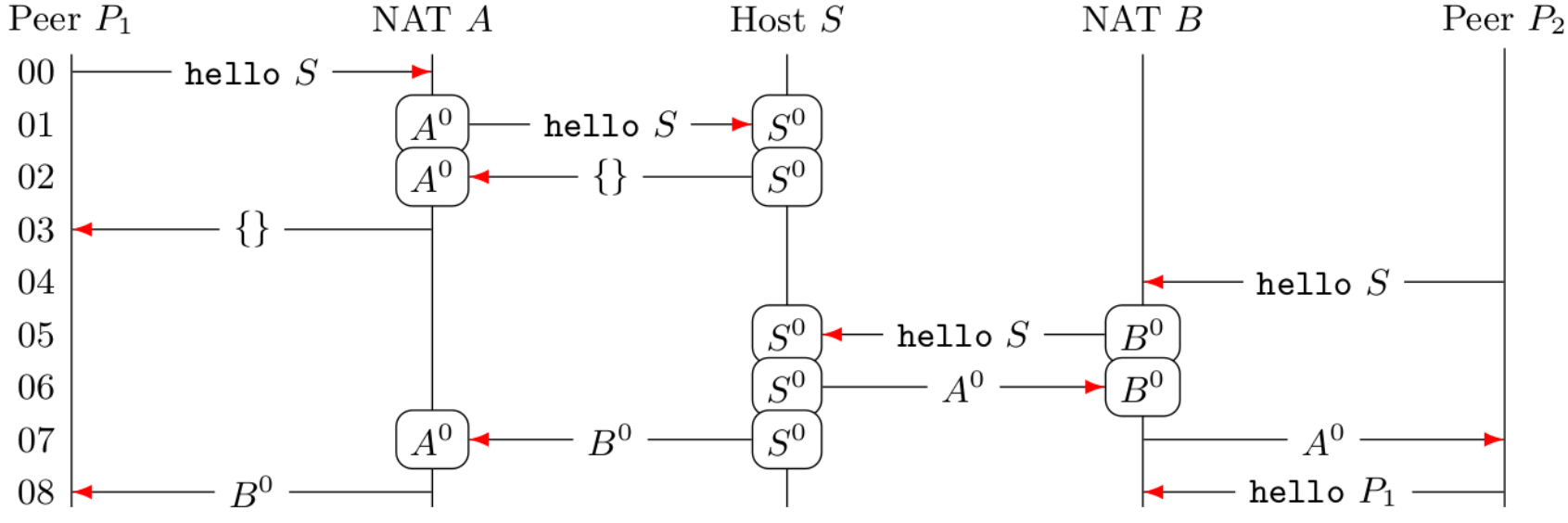
NAT TRAVERSAL

PEERS COMMUNICATION ISSUES



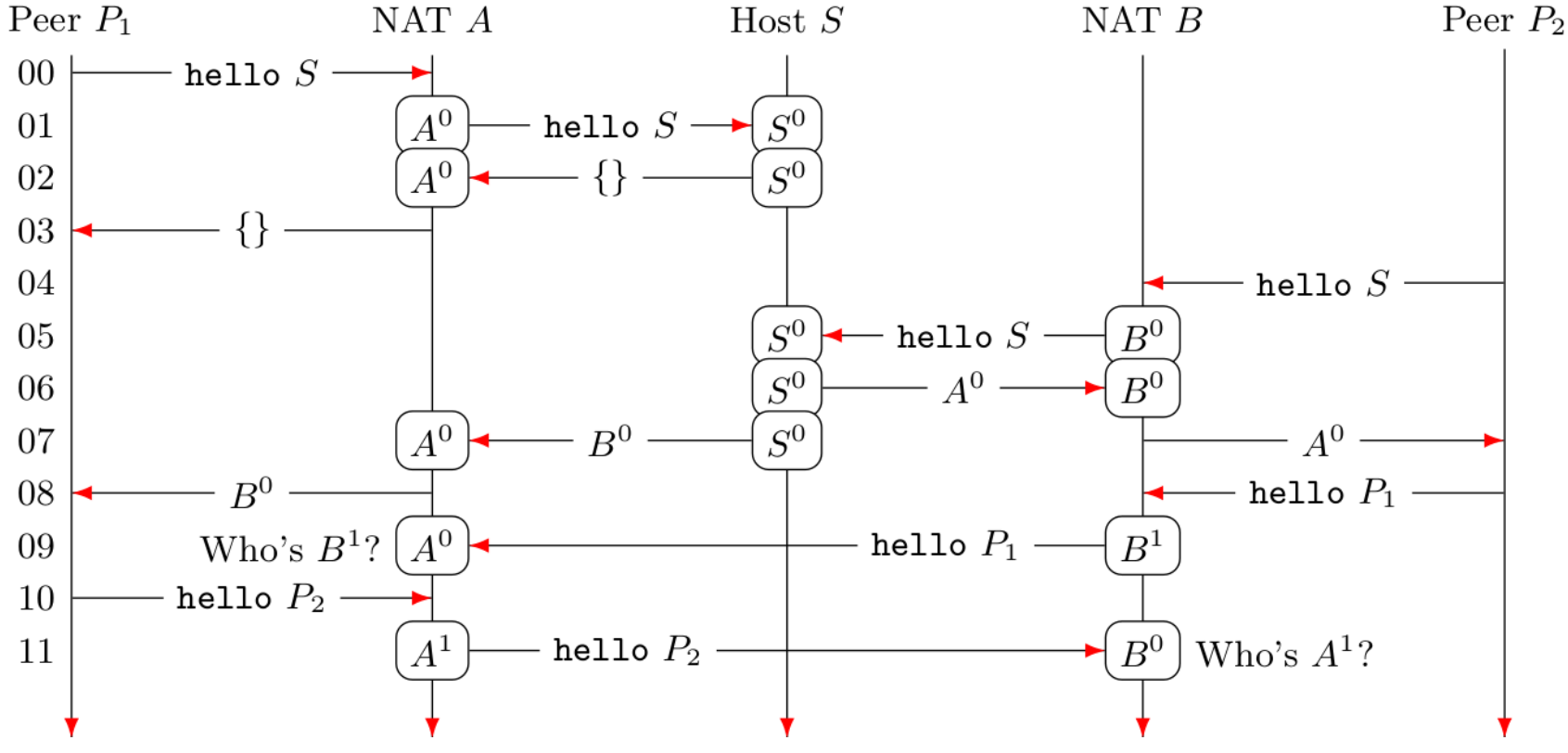
NAT TRAVERSAL

PEERS COMMUNICATION ISSUES



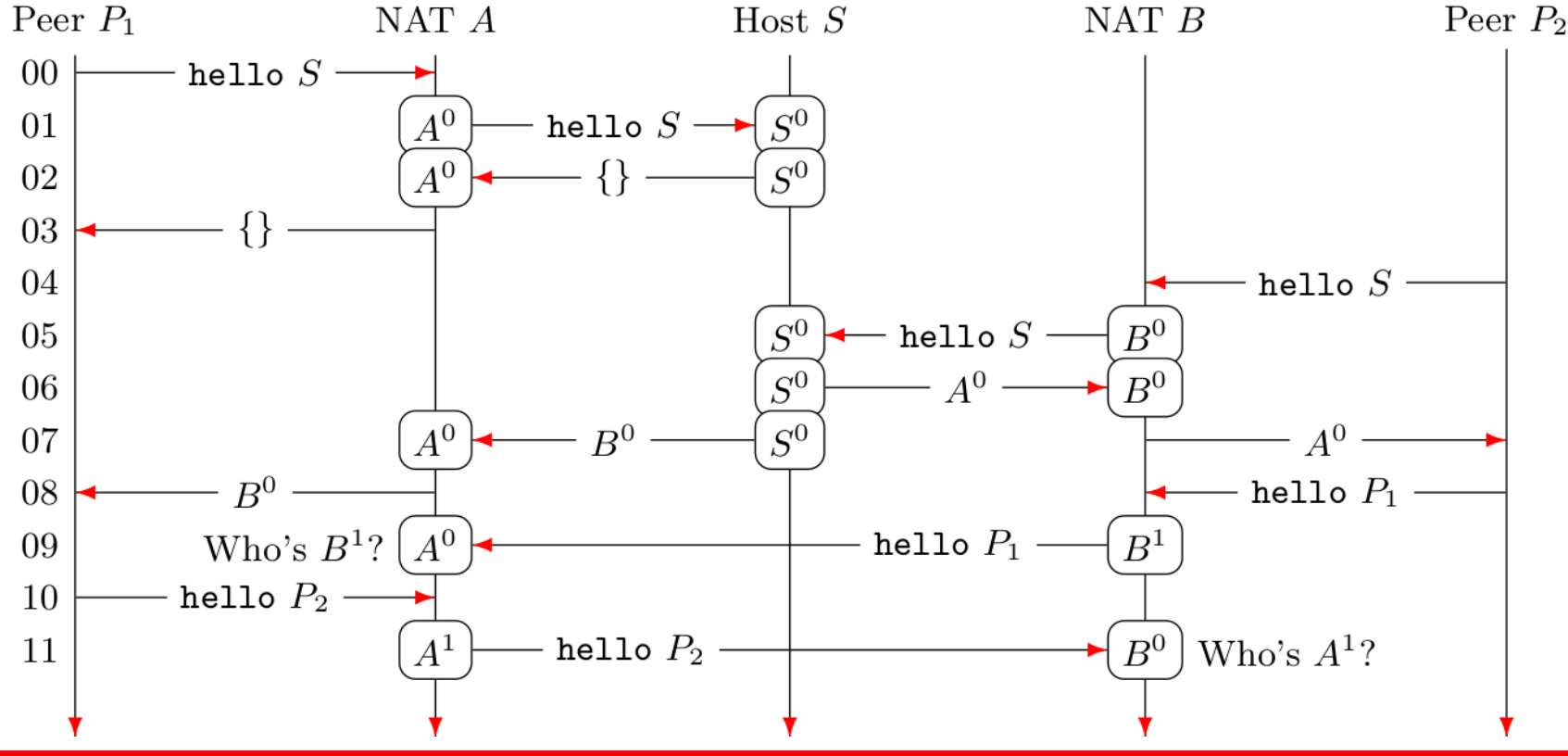
NAT TRAVERSAL

PEERS COMMUNICATION ISSUES



NAT TRAVERSAL

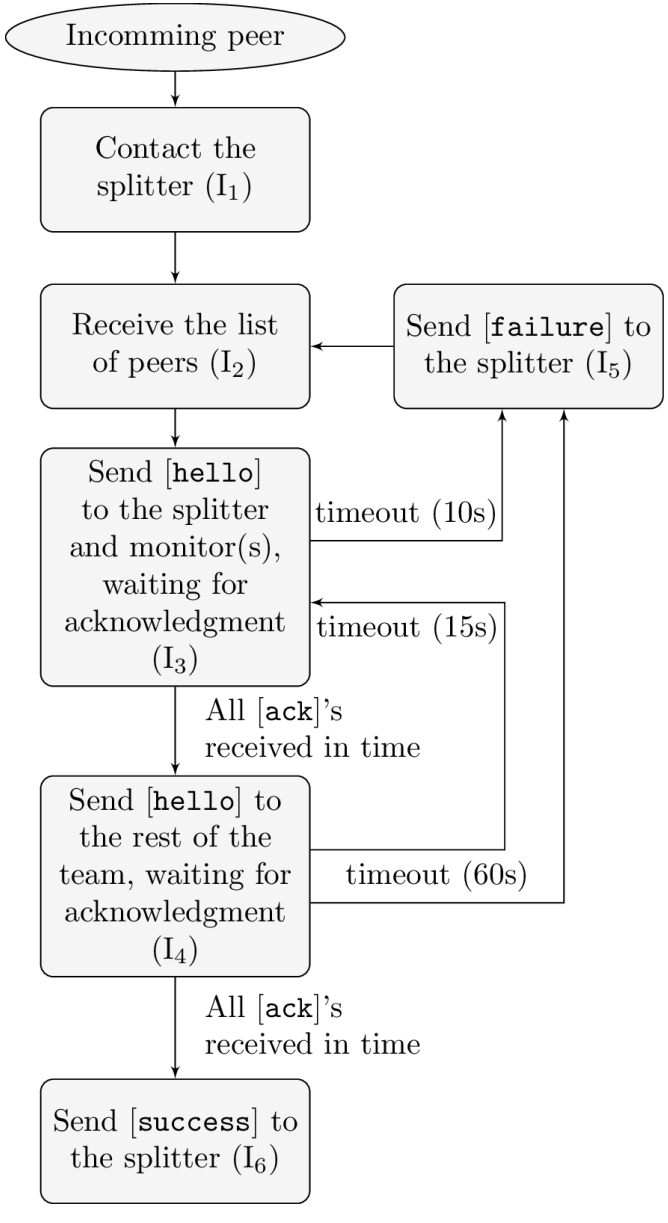
PEERS COMMUNICATION ISSUES



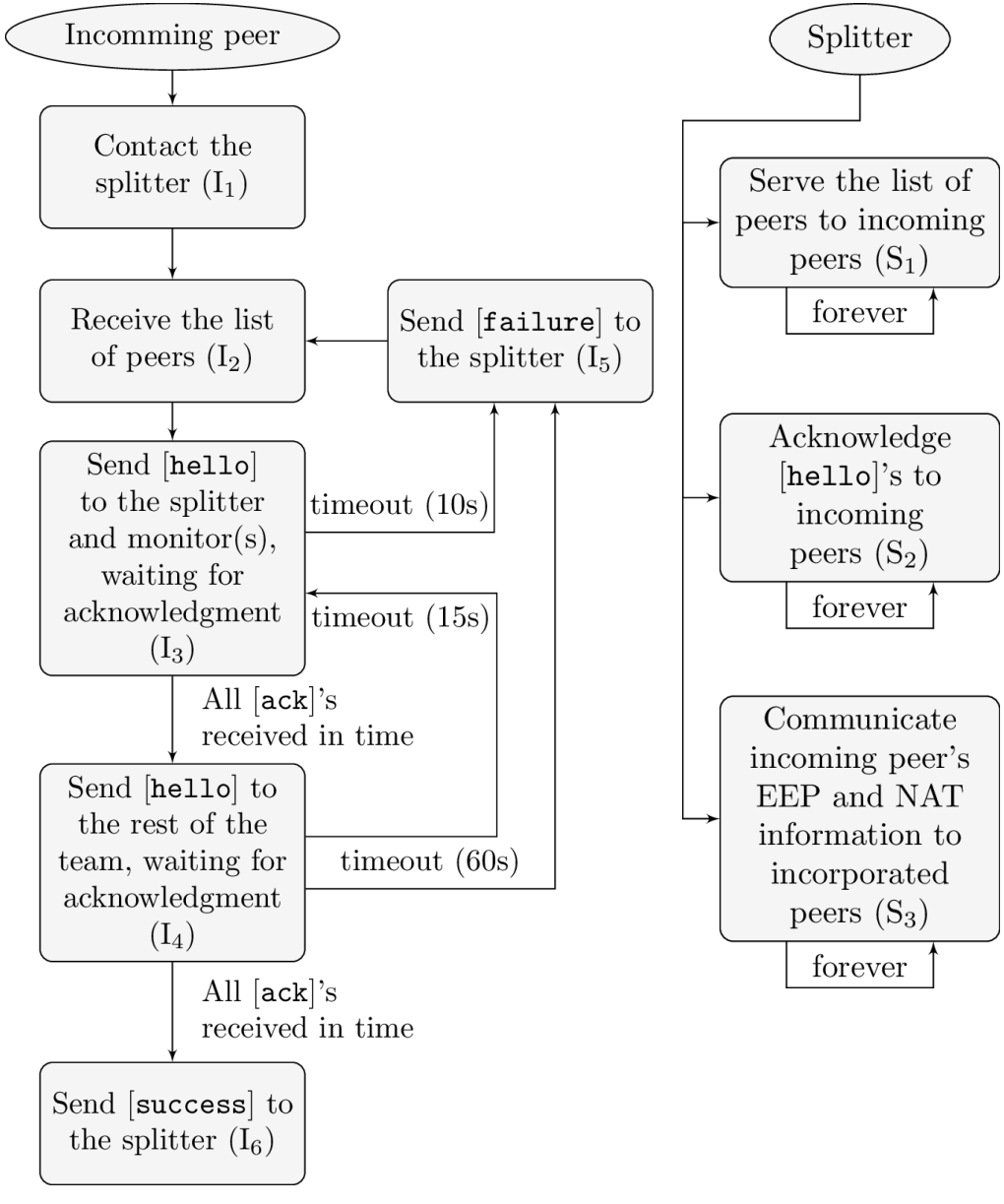
NAT device A does not have a translation entry for endpoint B_1, and NAT device B does not have a translation entry for endpoint A_1

COLLABORATIVE PORT PREDICTION

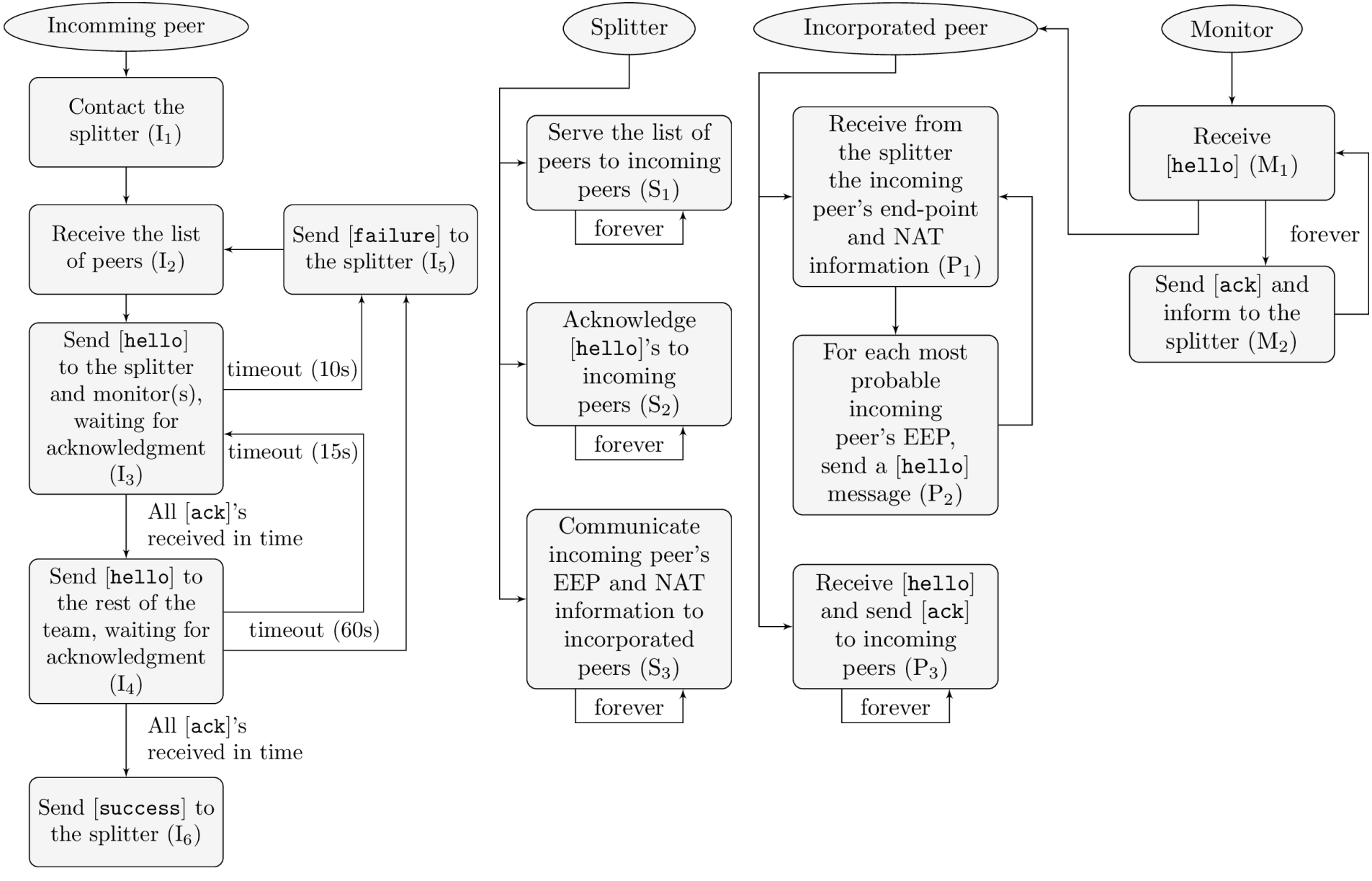
COLLABORATIVE PORT PREDICTION



COLLABORATIVE PORT PREDICTION

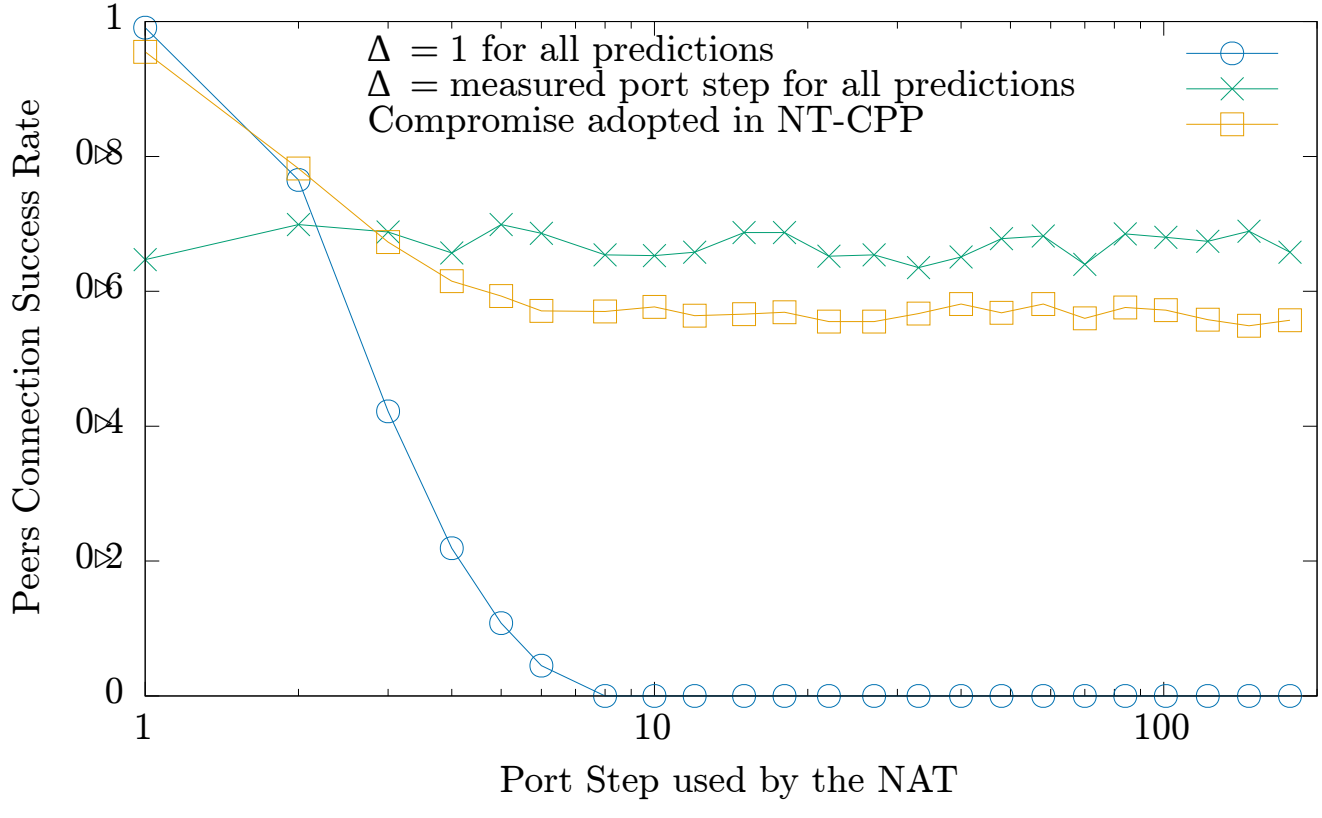


COLLABORATIVE PORT PREDICTION



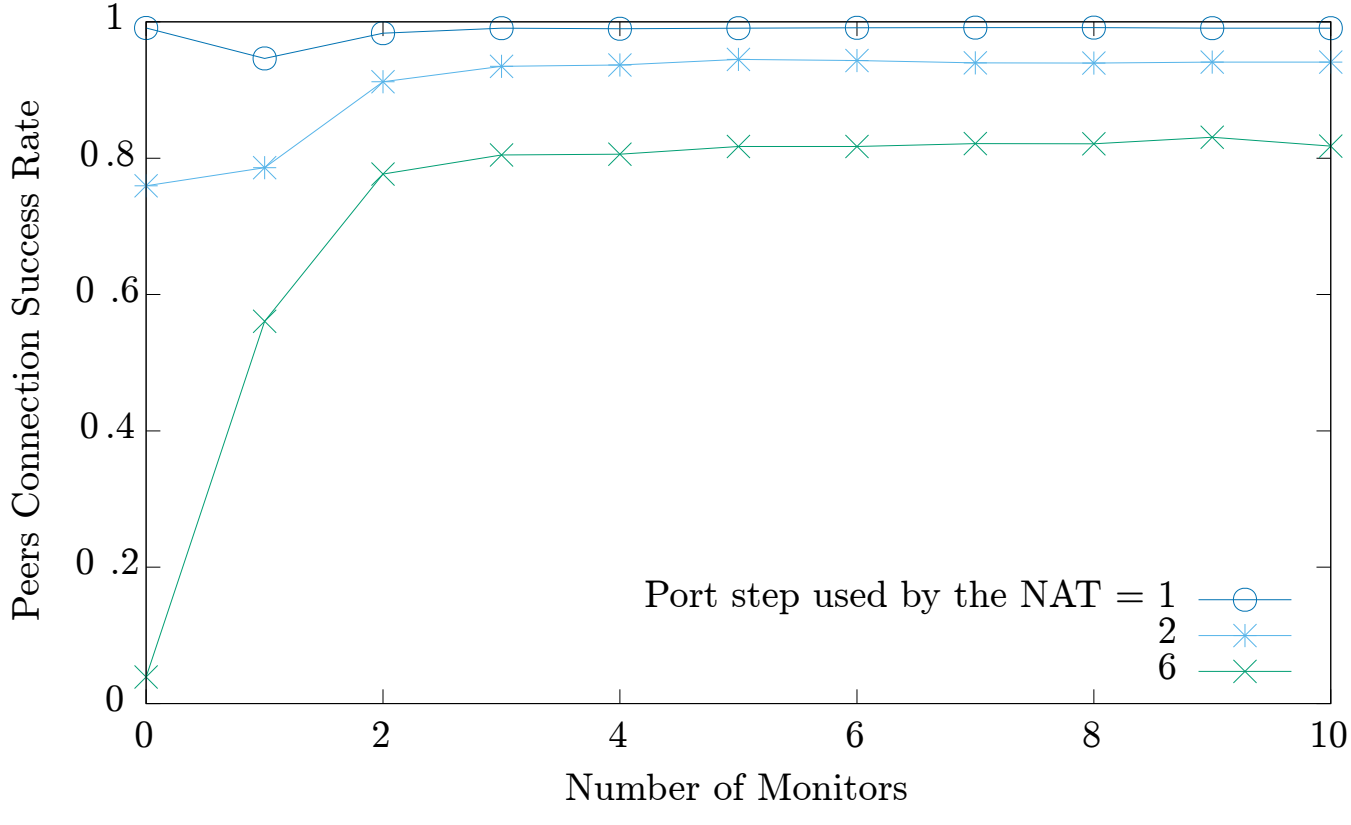
EXPERIMENTS

Port step: [1 to 200] | **Monitors:** 1 | Port predictions $n = 20$



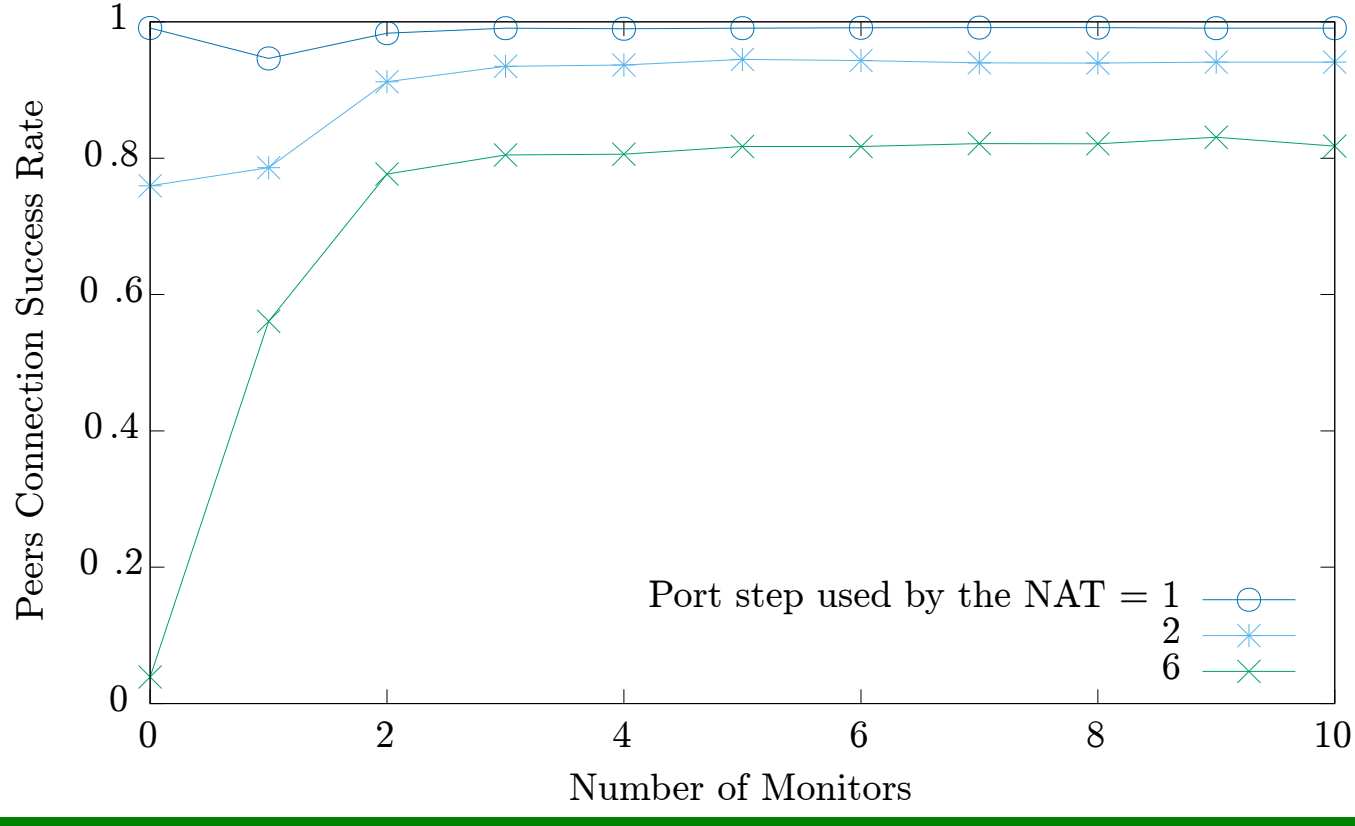
EXPERIMENTS

N : [1 to 10] | **Port step:** [1 to 6] | Port predictions $n = 20$



EXPERIMENTS

N : [1 to 10] | **Port step:** [1 to 6] | Port predictions $n = 20$



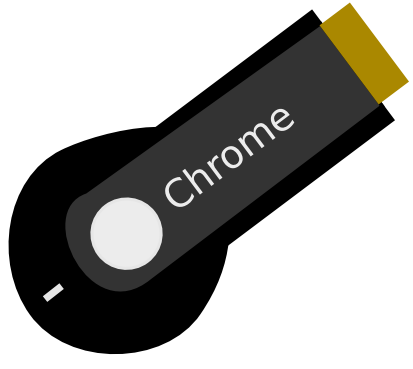
Only two monitor peers collaborating leads to higher traversal success rates

OUTLINE

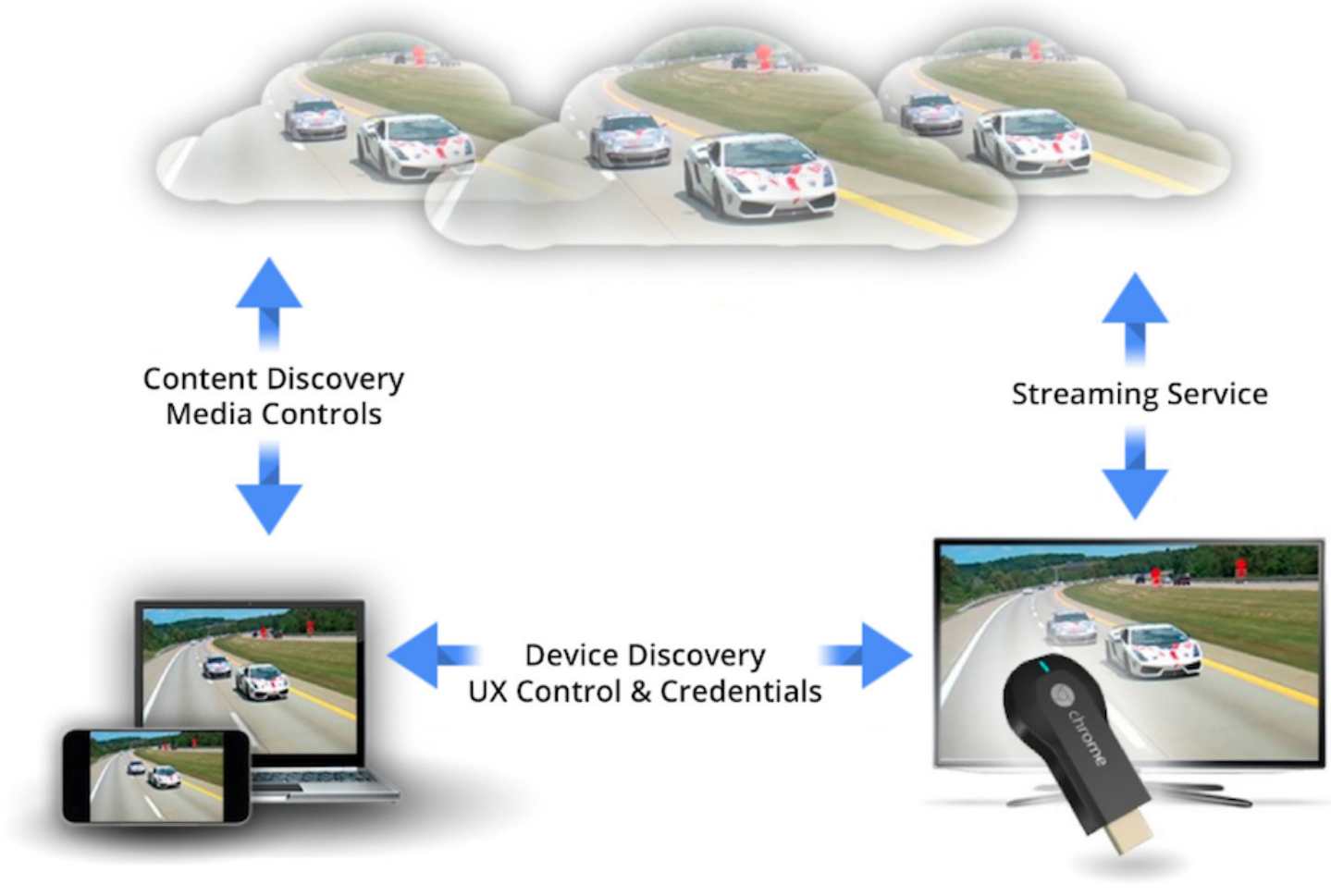
- INTRODUCTION
  - Background
  - Research Questions
- PEER TO PEER STRAIGHFORWARD PROTOCOL
  - An application-layer protocol that provides real-time broadcasting on the Internet
- SECURITY
  - Pollution Attacks
  - Strategies applied to Traditional Networks
  - Strategies applied to Software Defined Networks
- **EFFICIENCY**
  - NAT Traversal using Collaborative Port Prediction
  - Runing P2PSP on Embeded Devices
- CONCLUSIONS
  - Key Results and Future Work

GOOGLE CHROMECAST

Google Chromecast (GC) is a device designed as small dongle, which enable users with a mobile device or personal computer to play Internet-streamed audio-visual content on a high-definition television through mobile and web apps that support the Google Cast technology.



HOW DOES IT WORK?



BUILT-IN TECHNOLOGIES

BUILT-IN TECHNOLOGIES

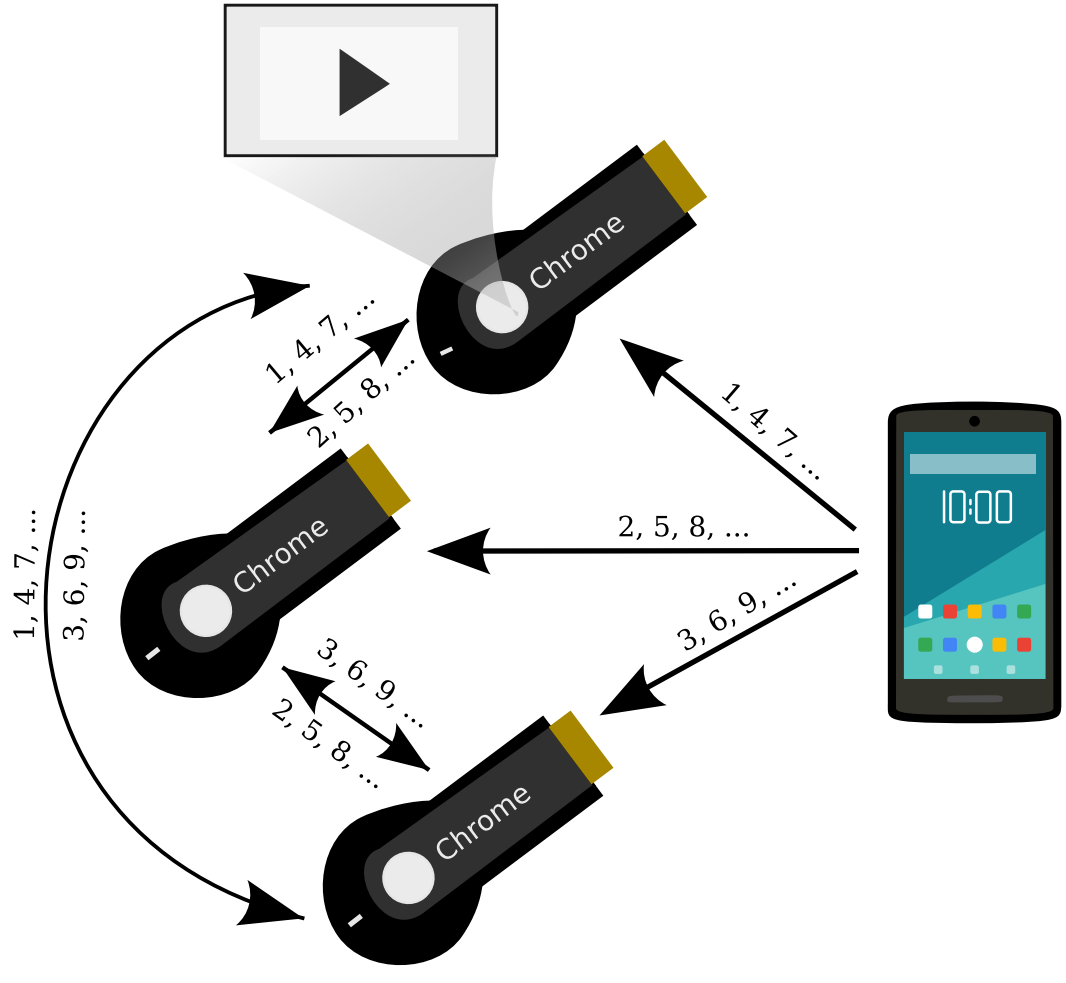
BUILT-IN TECHNOLOGIES

BUILT-IN TECHNOLOGIES

MSE

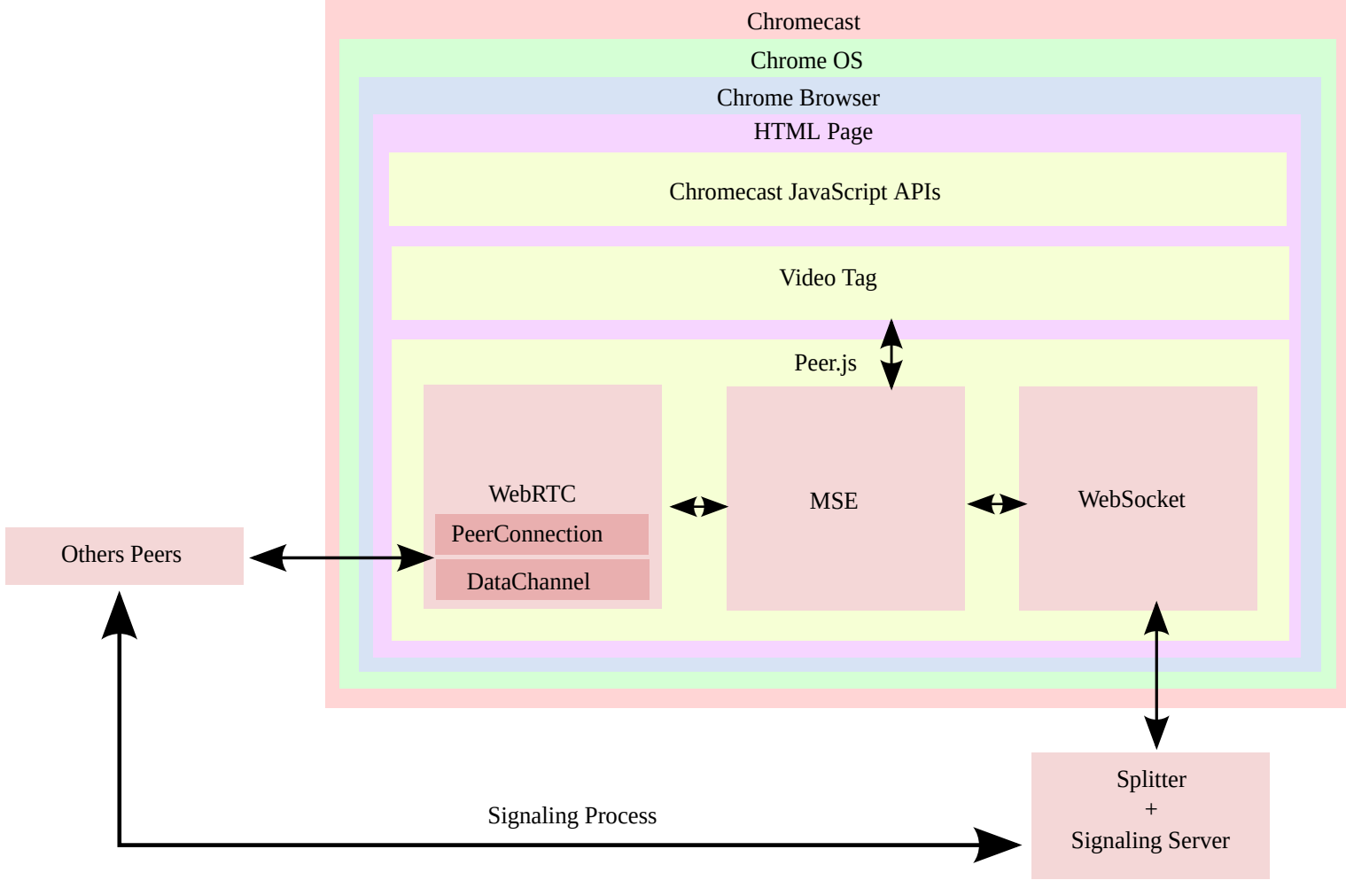
DEVELOPMENT

A P2PSP CHROMECAST TEAM



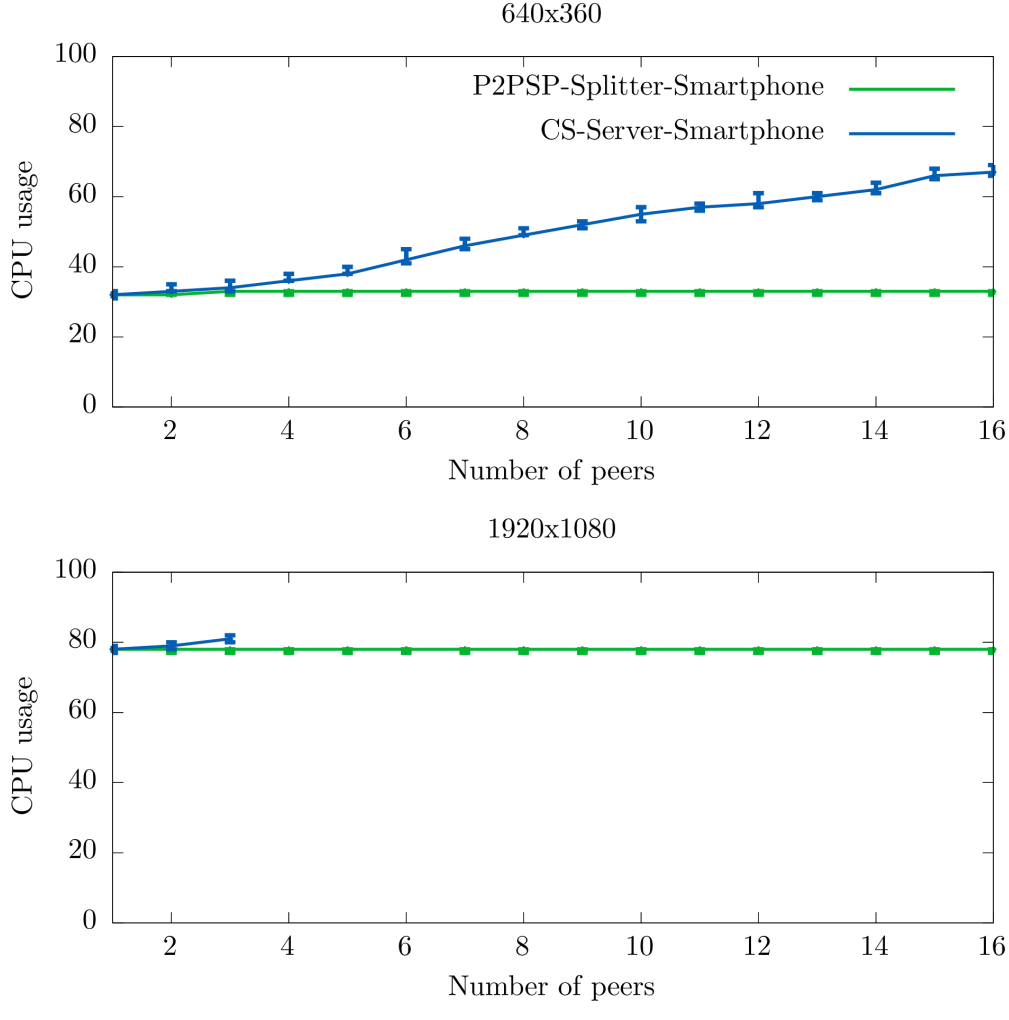
DEVELOPMENT

P2PSP CHROMECAST PEER ARCHITECTURE



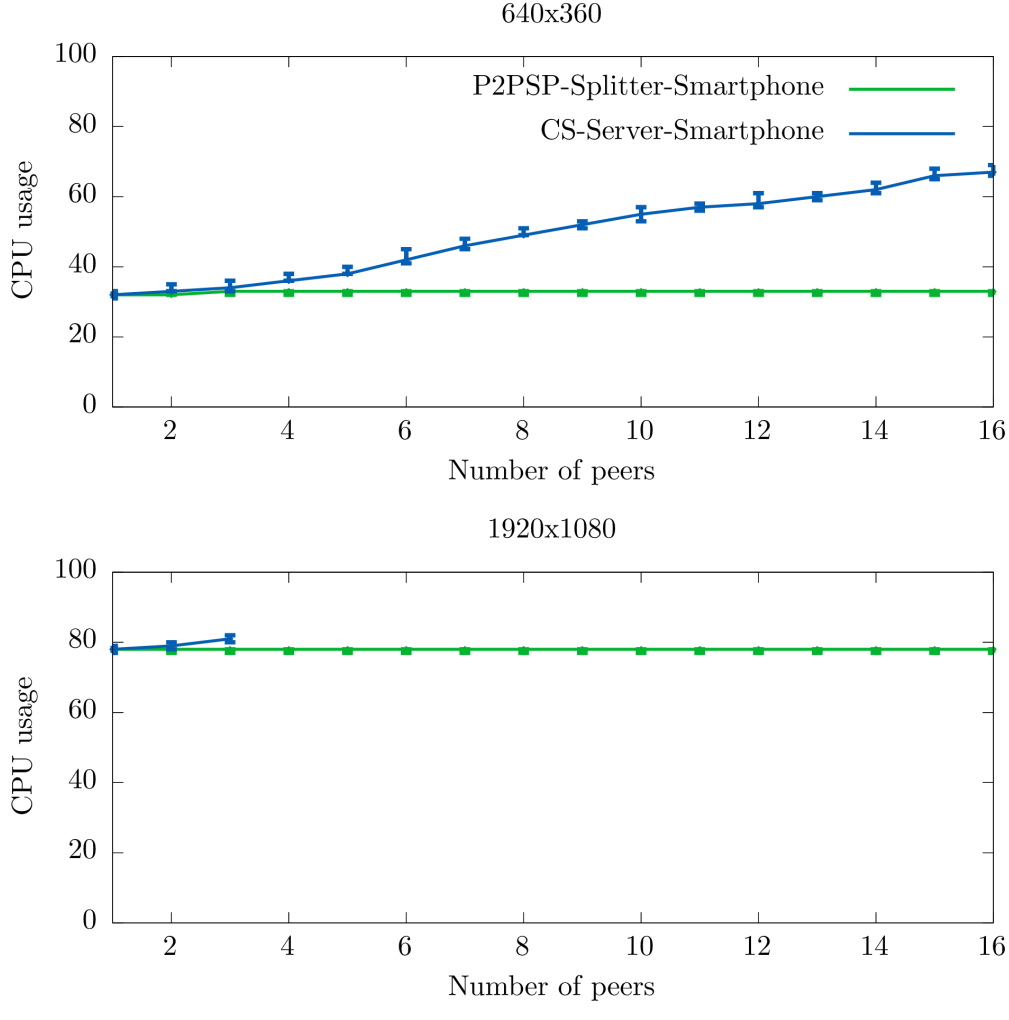
EXPERIMENTS

Streaming a MP4 (H.264 + AAC) video at 24 FPS. Resolutions: 640x360 and 1920x1080



EXPERIMENTS

Streaming a MP4 (H.264 + AAC) video at 24 FPS. Resolutions: 640x360 and 1920x1080

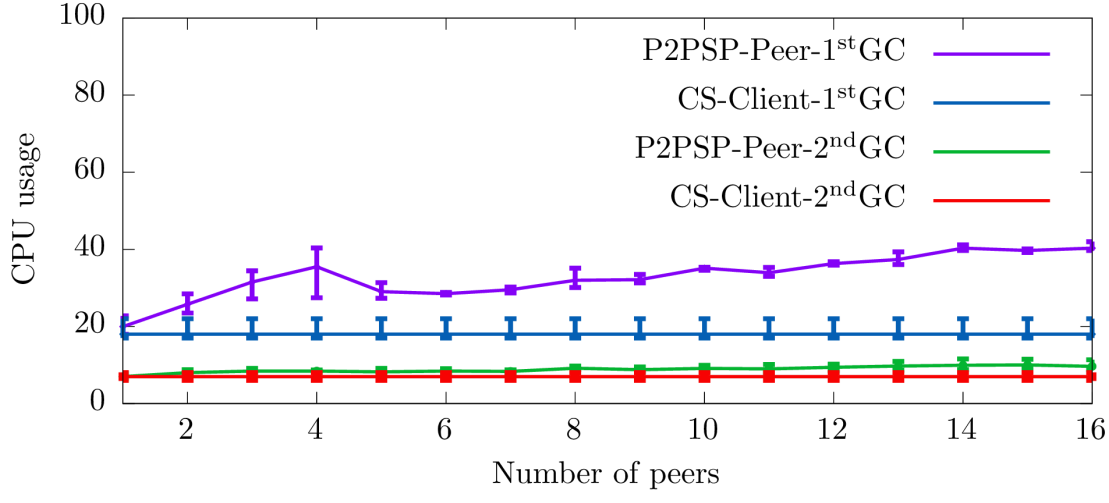


Number of peers receiving the media increases significantly in comparison with a CS model

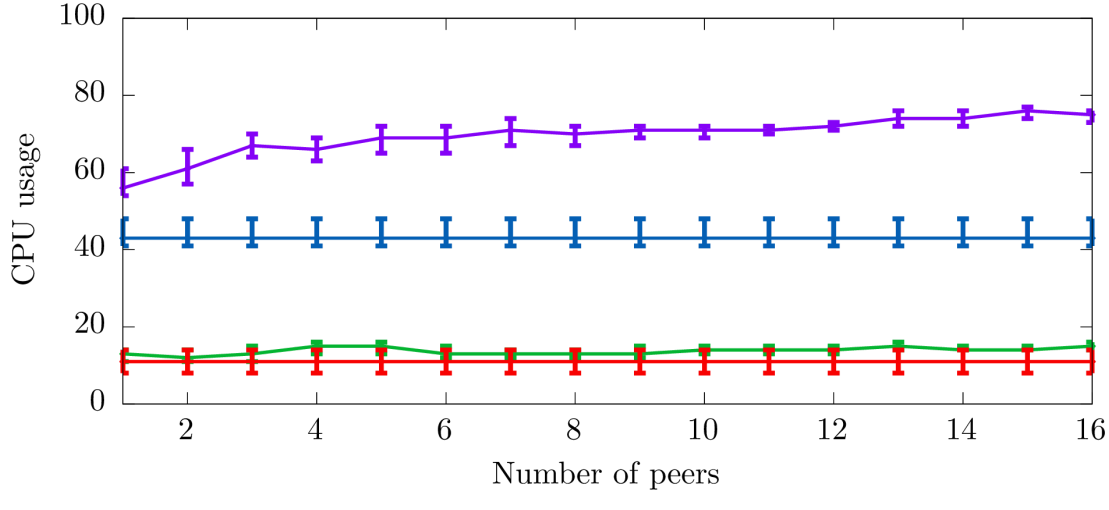
EXPERIMENTS

Streaming a MP4 (H.264 + AAC) video at 24 FPS. Resolutions: 640x360 and 1920x1080

640x360

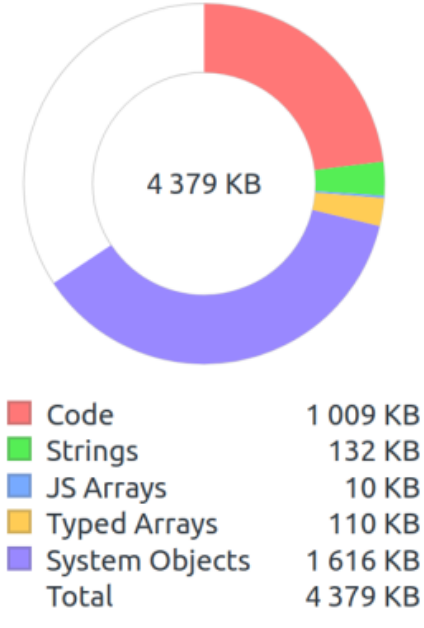


1920x1080

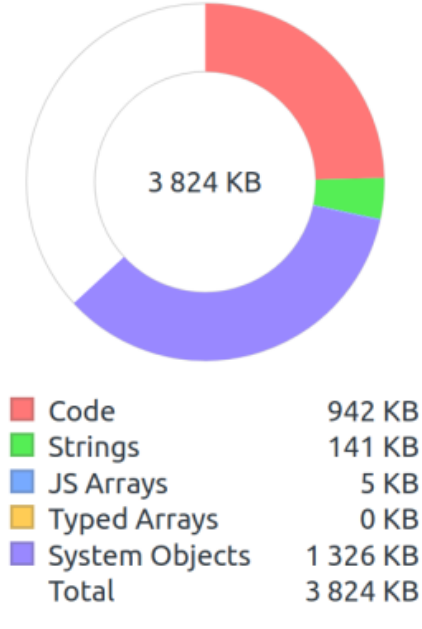


EXPERIMENTS

Streaming a MP4 (H.264 + AAC) video at 24 FPS. Resolutions: 640x360 and 1920x1080



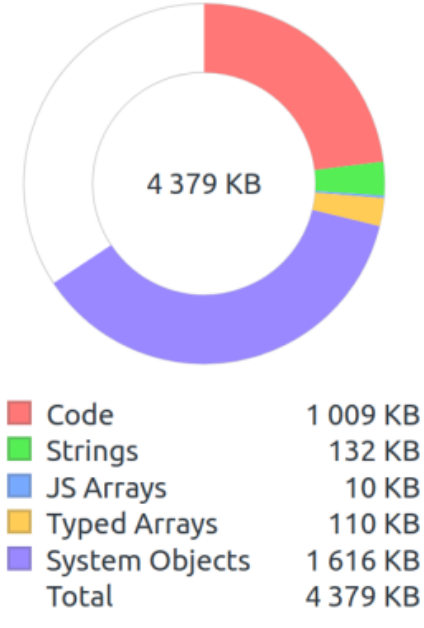
(a) P2PSP model.



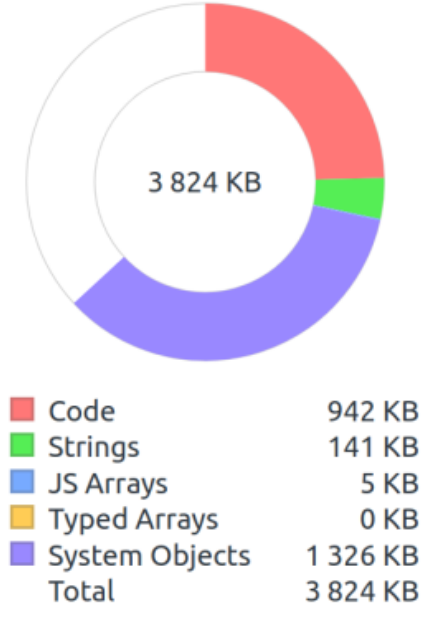
(b) CS model.

EXPERIMENTS

Streaming a MP4 (H.264 + AAC) video at 24 FPS. Resolutions: 640x360 and 1920x1080



(a) P2PSP model.



(b) CS model.

A peer can run on a GC improving the QoE of the user when connected to an HDMI TV

OUTLINE

- INTRODUCTION
  - Background
  - Research Questions
- PEER TO PEER STRAIGHFORWARD PROTOCOL
  - An application-layer protocol that provides real-time broadcasting on the Internet
- SECURITY
  - Pollution Attacks
  - Strategies applied to Traditional Networks
  - Strategies applied to Software Defined Networks
- EFFICIENCY
  - NAT Traversal using Collaborative Port Prediction
  - Runing P2PSP on Embeded Devices
- **CONCLUSIONS**
  - Key Results and Future Work

CONCLUSIONS

EFFICIENCY

CONCLUSIONS

EFFICIENCY

1. We propose a **straightforward P2P Protocol** to ensure an easy implementation, even in resource-constrained devices. Moreover, we identified the main issues and proposed a modular design.

CONCLUSIONS

EFFICIENCY

1. We propose a **straightforward P2P Protocol** to ensure an easy implementation, even in resource-constrained devices. Moreover, we identified the main issues and proposed a modular design.

2. We proposed a new and simple **NAT Traversal algorithm that uses Collaborative Port Prediction** in which a low number of source ports is needed for traversal succes.s

CONCLUSIONS

EFFICIENCY

1. We propose a **straightforward P2P Protocol** to ensure an easy implementation, even in resource-constrained devices. Moreover, we identified the main issues and proposed a modular design.

2. We proposed a new and simple **NAT Traversal algorithm that uses Collaborative Port Prediction** in which a low number of source ports is needed for traversal succes.s

3. We got a implementation of the **P2PSP running on embedded devices** aiming of avoiding third-party media servers, in order to decrease the cost of the streaming and increase the degree of privacy.

CONCLUSIONS

SECURITY

CONCLUSIONS

SECURITY

1. Our experimental results show that **using only trusted peers** as a defense strategy is not appealing when the number of malicious peers can be large.

CONCLUSIONS

SECURITY

1. Our experimental results show that **using only trusted peers** as a defense strategy is not appealing when the number of malicious peers can be large.

2. The most severe possible **attack is fully mitigated**. For the remaining attacks, we can improve effectiveness to face them increasing the number of TPs.

CONCLUSIONS

SECURITY

1. Our experimental results show that **using only trusted peers** as a defense strategy is not appealing when the number of malicious peers can be large.

2. The most severe possible **attack is fully mitigated**. For the remaining attacks, we can improve effectiveness to face them increasing the number of TPs.

3. Deterministic quick MP detection, not only in pure SDN environments but also in mixed environments where some peers are on the Internet and others are under managed networks.

PUBLICATIONS

INTERNATIONAL JOURNALS (JCR)

1. Medina-Lopez, C., Mertens, M.B., Gonzalez-Ruiz, V. & Casado, L.G. (2019). Reducing streaming cost while increasing privacy: A case study on a smartphone and chromecast using peer-to-peer technology to skip third-party servers. IEEE Consumer Electronics Magazine, DOI: 10.1109/MCE.2018.2880810. 8, 50–55 Impact factor JCR 2018: 3.273. **Journal ranking: 10/52 (Q1) in Computer science, Hardware & Architecture.** 76/265 (Q2) in Engineering, Electrical & Electronic. 29/88 (Q2) in Telecommunications.

2. (Under Review) Medina-Lopez, C., Casado, L.G, Yuansong Qiao & Gonzalez-Ruiz, V. An SDN Approach to Detect Targeted Attacks in P2P Fully Connected Overlays. International Journal of Information Security. **Impact Factor JCR 2018: 1.822. 44/107 (Q2) in Computer science, Software Engineering.** 42/104 (Q2) in Computer science, Theory and Methods. 92/155 (Q3) Computer science, Information systems.

3. (Under Review) Medina-Lopez, C., García-Ortiz J.P, Martinez, J.A, Casado, L.G, Gonzalez-Ruiz, V. NAT Traversal in P2P Networks using Collaborative Port Prediction. Peer-to-Peer Networking and Applications. **Impact Factor JCR 2018: 2.397. 70/155 (Q2) Computer science, Information systems.** 41/88 (Q2) Telecommunications.

PUBLICATIONS

INTERNATIONAL CONFERENCES WITH DOI

1. Medina-López, C., González-Ruiz, V. & Casado, L. (2017). On mitigating pollution and free-riding attacks by Shamir's Secret Sharing in fully connected P2P systems. In Wireless Communications and Mobile Computing Conference (IWCMC), 2017 13th International, 711–716, DOI: 10.1109/IWCMC.2017.7986372. IEEE.**GGS Rating: B, GGS Class: 3**.

2. Medina-López, C., Shakirov, I., Casado, L. & González-Ruiz, V. (2017). On pollution attacks in fully connected P2P networks using trusted peers. In Intelligent Systems Design and Applications, 144–153, DOI: 10.1007/978-3-319-53480-0. Springer, Cham, Porto. **CORE 2016: rank C.**

3. Medina-López, C., Casado, L. & González-Ruiz, V. (2015). Pollution Attacks Detection in the P2PSP Live Streaming System. In International Joint Conference. CISIS 2015. Advances in Intelligent Systems and Computing, 401–410, DOI: 10.1007/978-3-319-19713-5 34. Springer International Publishing. **CORE 2014: rank B.**

PUBLICATIONS

OTHER INTERNATIONAL CONFERENCES

1. Medina-López, C. (2015). Participation as a speaker. In GSoC 2015 Lightning Talks, **Google Inc., SunnyVale, CA. USA**

2. Medina-López, C., García Ortiz, J.P., Naranjo, J., Casado, L. & González-Ruiz, V. (2014). IPTV using P2PSP and HTML5+WebRTC. **In 4th W3C Web and TV Workshop**, 5, IRT, W3C, Munchen, Germany

PUBLICATIONS

NATIONAL CONFERENCES

1. Medina-López, C., González-Ruiz, V., Casado, L.G., Naranjo, J. & García-Ortiz, J.P. (2015). Ejecutando peers p2psp en google chromecast. In Actas VI Jornadas de Computación Empotrada, 123–129, Cordoba

2. Medina-López, C., Naranjo, J., García-Ortiz, J.P., Casado, L.G. & González-Ruiz, V. (2013). Execution of the P2PSP protocol in parallel environments. In G.B. y Alberto A. Del Barrio Garcia, ed., Actas XXIV Jornadas de Paralelismo, 216–221, Madrid

PUBLICATIONS

OTHER PUBLICATIONS PRODUCED DURING THE ELABORATION OF THIS THESIS

1. Andujar, A. & Medina-López, C. (2019). Exploring New Ways of eTandem and Telecollaboration Through the WebRTC Protocol: Students' Engagement and Perceptions. International Journal of Emerging Technologies in Learning (iJET), 14, 200–217

2. Medina-Lopez, C., Casado, L.G. & Gonzalez-Ruiz, V. (2015). P2PSP: un protocolo de red sencillo como herramienta para el aprendizaje basado en proyectos. Experiencias Docentes en Redes de Computadores, 1, 35–41

THANKS!

Google Summer of Code