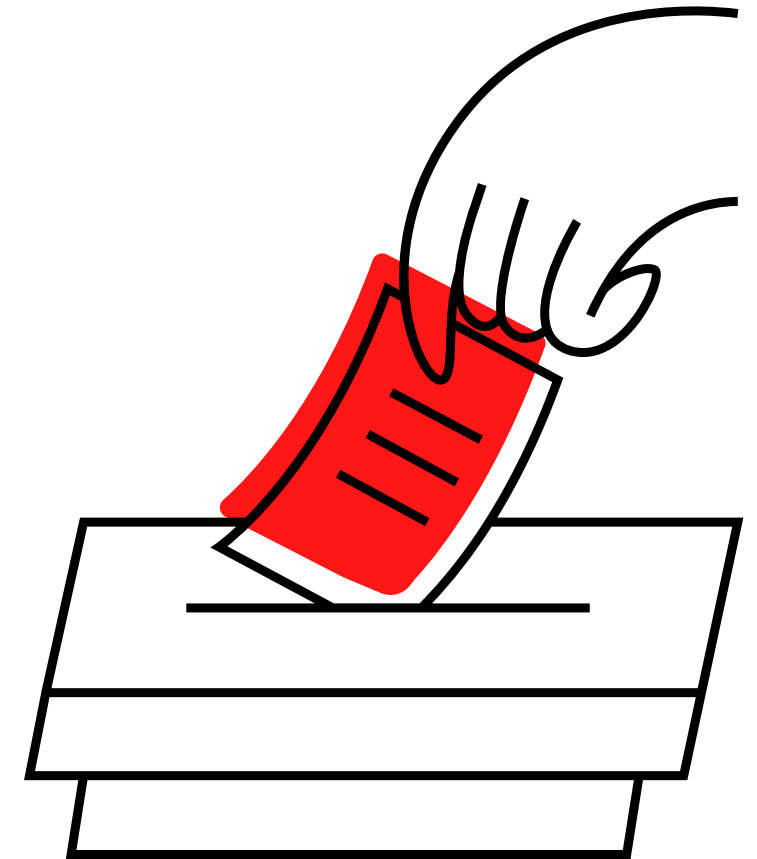
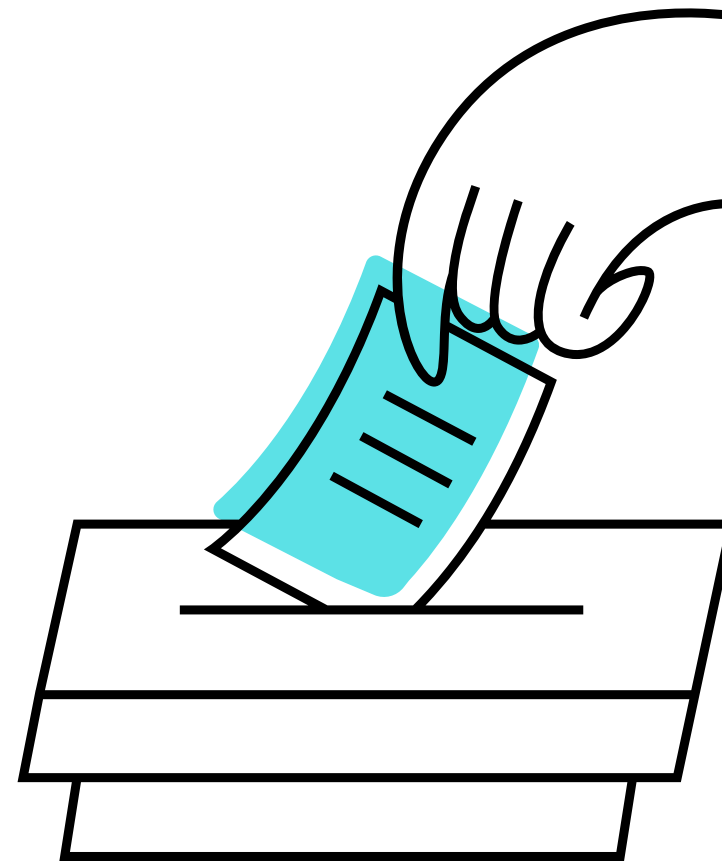
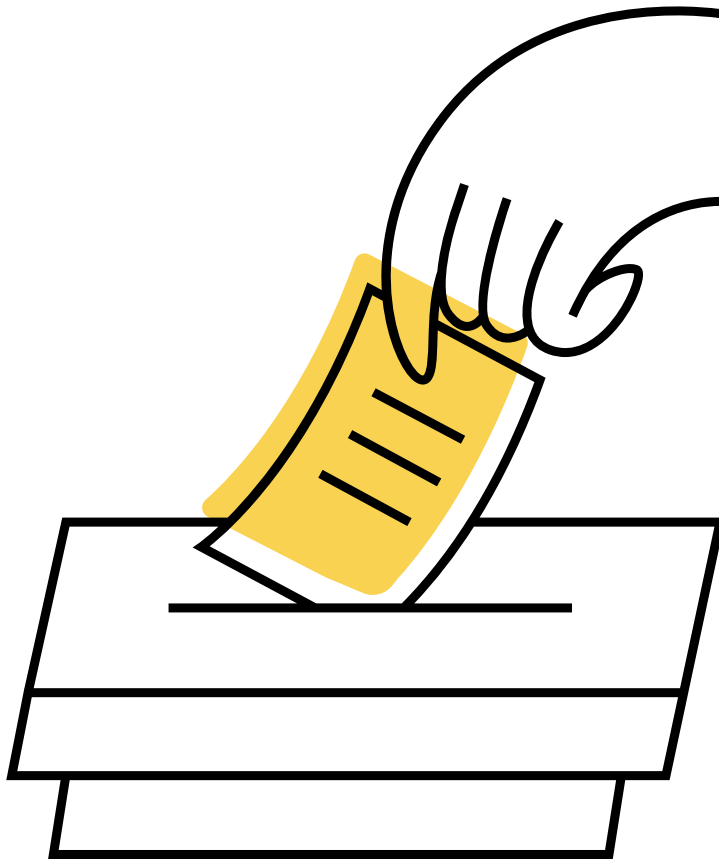
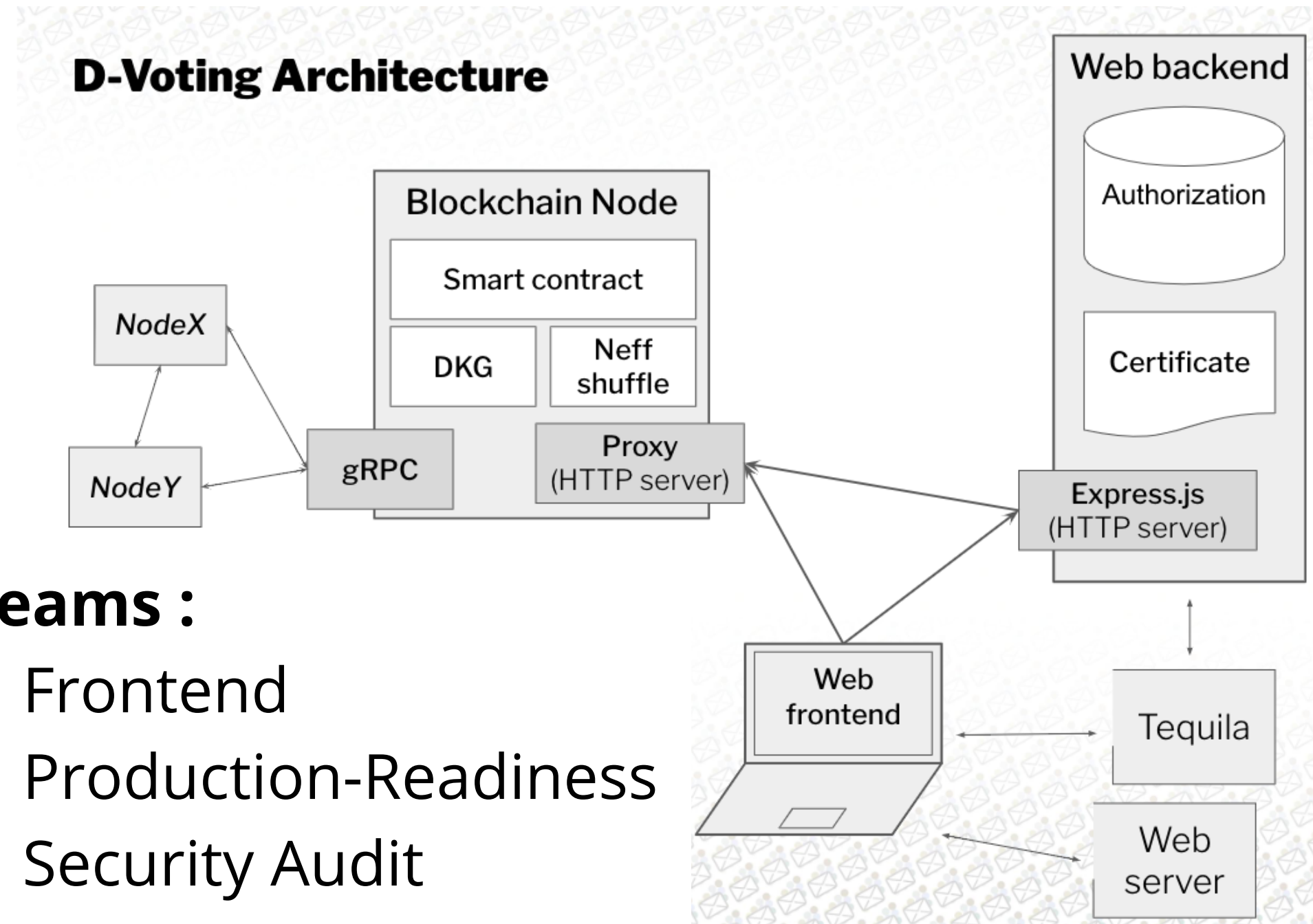


D-voting

Final Presentation



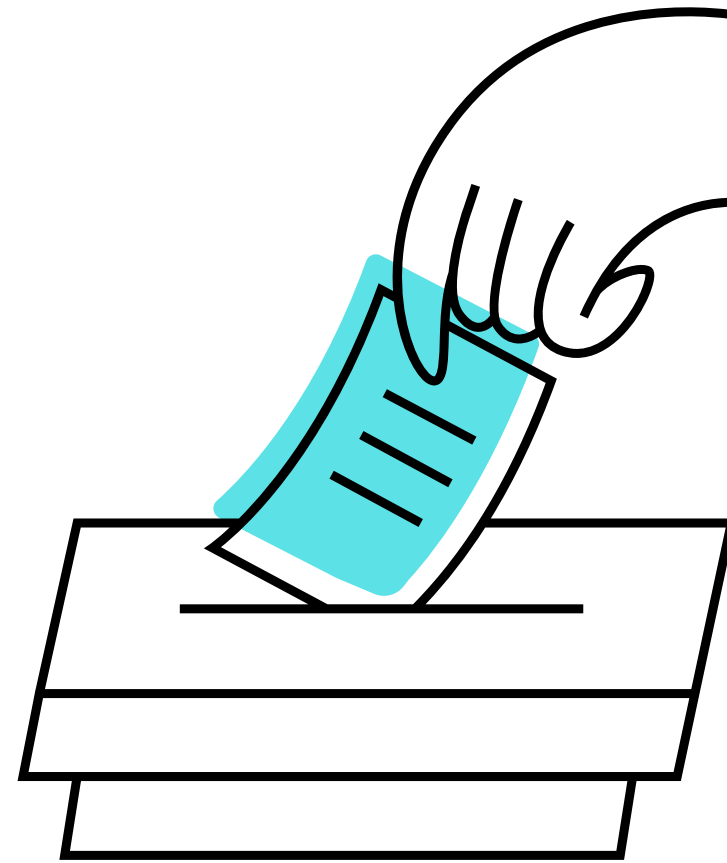
What is D-Voting?



3 teams :

- Frontend
- Production-Readiness
- Security Audit

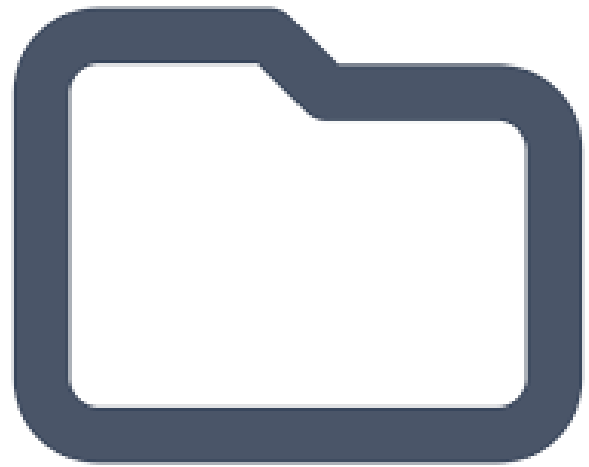
Front-end



Ahmed Elalamy, Ghita Tagemouati, Khadija Tagemouati

Introduction

Form



Subject



Rank



Select



Text

Goals

Goals

Usability

Goals

Usability

Hint Individual results

Goals

Usability

Robustness

Goals

Usability

Robustness

Authorization mechanism

Goals

Usability

Robustness

Accessibility

Goals

Usability

Robustness

Accessibility

Internationalization

Features & Implementations

Authorization Mechanism



Authorization Mechanism

BEFORE

```
export const enum UserRole {  
  Admin = 'admin',  
  Operator = 'operator',  
  Voter = 'voter',  
}
```

How to make sure that only the form creator can manage it ?

Authorization Mechanism



Roles

Authorization Mechanism

BEFORE

AFTER

Roles



Policies

Authorization Mechanism

A red, rectangular stamp with a distressed, ink-like texture. The word "AFTER" is written in bold, uppercase letters inside the stamp, which is tilted slightly to the right.

```
p, "SCIPER", "subject", "action"
```

Authorization Mechanism

AFTER



Authorization Mechanism

A red, rectangular stamp with a distressed, ink-like texture. The word "AFTER" is written in bold, white, sans-serif capital letters, tilted slightly upwards to the right.

```
p, 330361, roles, list  
p, 330361, election, create  
p, 330361, roles, remove  
p, 330361, roles, add  
p, 330361, proxies, post  
p, 330361, proxies, put  
p, 330361, proxies, delete
```

```
p, 175129, roles, list  
p, 175129, election, create  
p, 175129, roles, remove  
p, 175129, roles, add  
p, 175129, proxies, post  
p, 175129, proxies, put  
p, 175129, proxies, delete
```

Authorization Mechanism

AFTER

```
function isAuthorized(sciper: number | undefined, subject: string, action: string): boolean {  
  return enf.enforceSync(sciper, subject, action);  
}
```

Authorization Mechanism



More than *sixty* lines !

Authorization Mechanism



id character varying (256) 🔒	ptype character varying (256) 🔒	v0 character varying (256) 🔒	v1 character varying (256)	v2 character varying (256) 🔒
	p	330383	roles	list
	p	330383	roles	remove
	p	330383	roles	add
	p	330383	proxies	post
	p	330383	proxies	put
	p	330383	proxies	delete
	p	330383	election	create
	p	330382	roles	list
	p	330382	roles	remove
	p	330382	roles	add
	p	330382	proxies	post

Authorization Mechanism



How to make sure that only the form creator can manage it ?

Authorization Mechanism

AFTER

```
app.put('/api/evoing/authorizations', (req, res) => {  
  if (!isAuthorized(req.session.userid, SUBJECT_ELECTION, ACTION_CREATE)) {  
    res.status(400).send('Unauthorized');  
    return;  
  }  
  const { FormID } = req.body;  
  enf.addPolicy(String(req.session.userid), FormID, ACTION_OWN);  
});
```


Translation & internationalization

Extension of Front-end
functionalities

Future Improvements

Future Improvements

- Improve programmers' experience

Future Improvements

- Improve programmers' experience
- Improve error handling

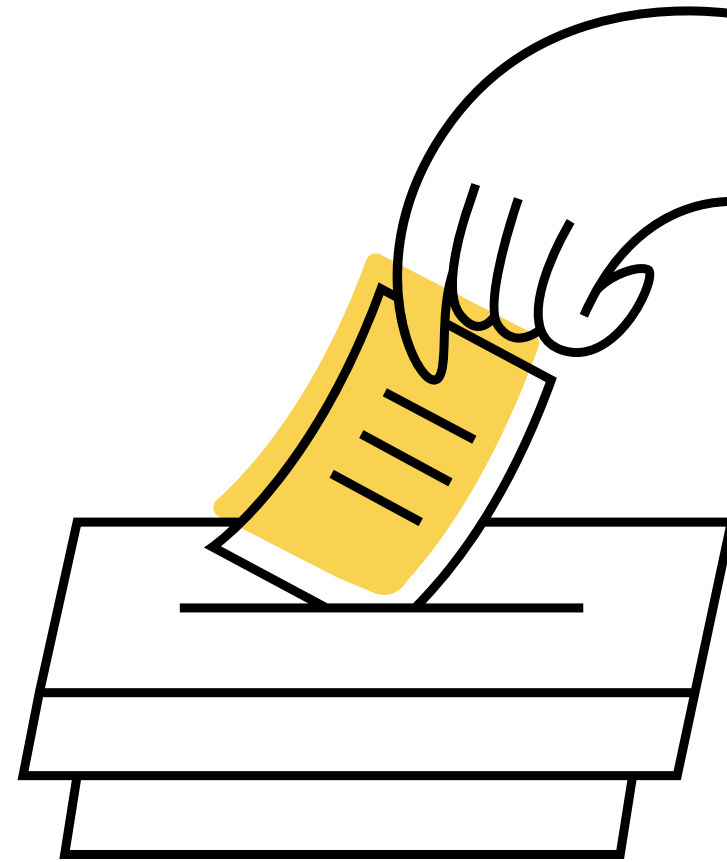
Future Improvements

- Improve programmers' experience
- Improve error handling
- Use a new authorization model

Future Improvements

- Improve programmers' experience
- Improve error handling
- Use a new authorization model
- Make individual results only available to admins

Production Readiness



Amine Benaziz, Albert Troussard

Production readiness

Pre-Project Observations

- Lack of testing in realistic conditions
- Need for refactoring
- Lack of documentation
- Need for more metrics

Production readiness **Goals**

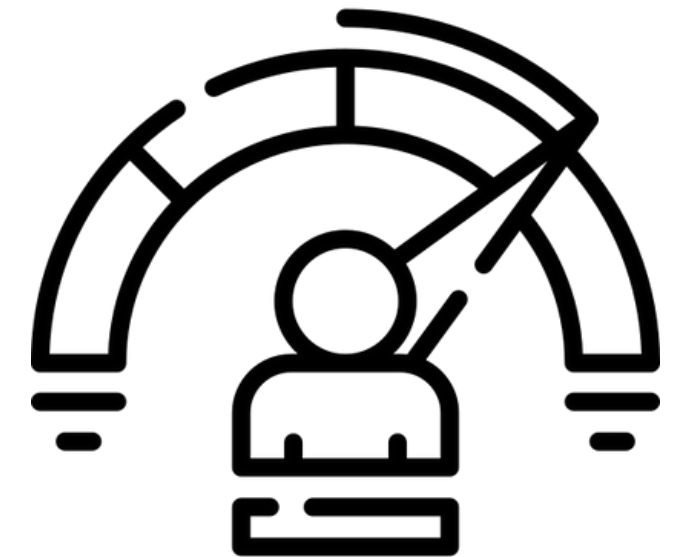
Usability

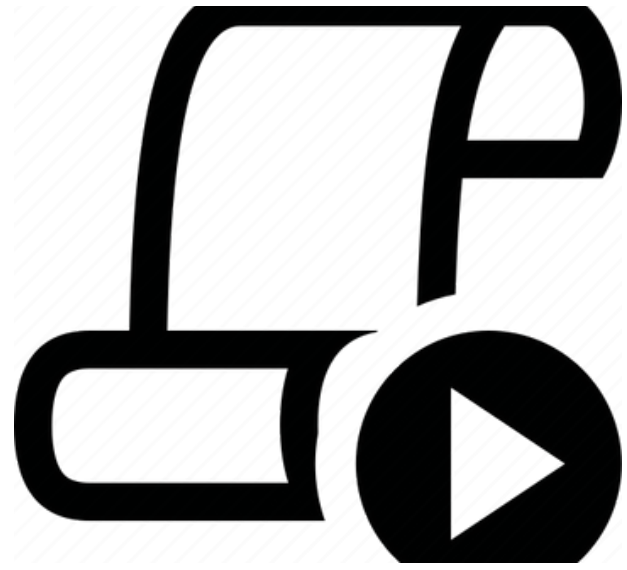


Robustness



Performance





Production readiness

Usability

Running script

- Script to quickly & easily set up the system
- Minimized complexity & automated setup steps
- Easily maintain the system & make changes as needed



Production readiness

Usability

Code metrics

- SonarCloud: Code Analysis Platform
- Code Coverage
- Complexity
- Bugs
- Security Vulnerabilities
- Identify areas that need testing & improvement



Production readiness

Robustness

Codebase quality.

- Review code & identify areas for improvement
- Refactor code & reduce code smells
- Improve codebase quality & ensure system security and reliability

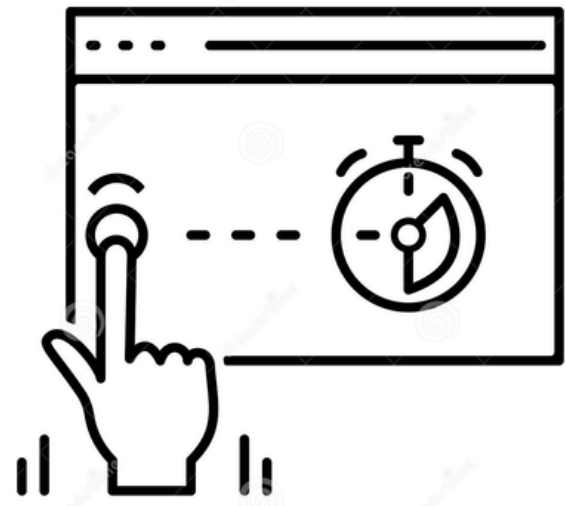


Production readiness

Robustness

Authorization Mechanism

- Fail-safe default mechanism based on Sciper numbers
- Give rights to other users (admins/operators)
- Improve security & ensure only authorized users have access



Production readiness

Robustness

Unit Testing

- Verify system functioning correctly
- Focus on backend of the system
- Ensure system security & reliability



Production readiness

Robustness

Integration Testing

- Ensure system functioning correctly and components working together
- Test different scenarios (crashing node, revote, null ballots)
- Identify potential issues before they become serious problems



Production readiness

Robustness

Scenario Test

- Simulates real-world election from start to finish
- Benchmarking system performance and reliability

Production readiness

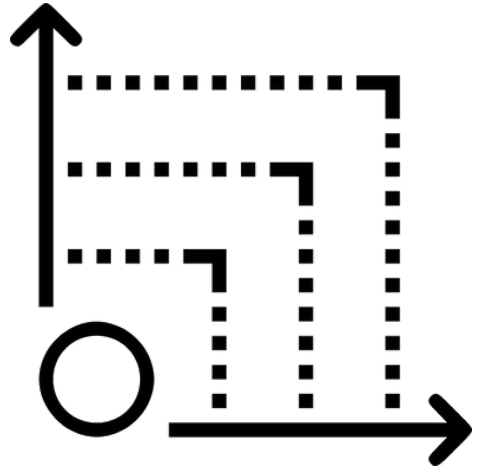
Robustness

- Time spent on each part of election aligned with expectations
- Main flaw of scenario test is lack of realism - txn only sent when previous accepted, takes time
- Needed test to mimic real-life, when ballots can be sent to blockchain at any time

Create form	Setup DKG	Open form	Casting 100 ballots
1.08s	3.56s	1.16s	138.39s

Close form	Shuffle ballots	Decrypt ballots
1.56s	31.3s	7.9s

- Simulation of sending 100 ballots to 10 nodes



Production readiness **Performance** Load testing

- Similar to the scenario test but more realistic.
- Difference is in the sending of transactions.

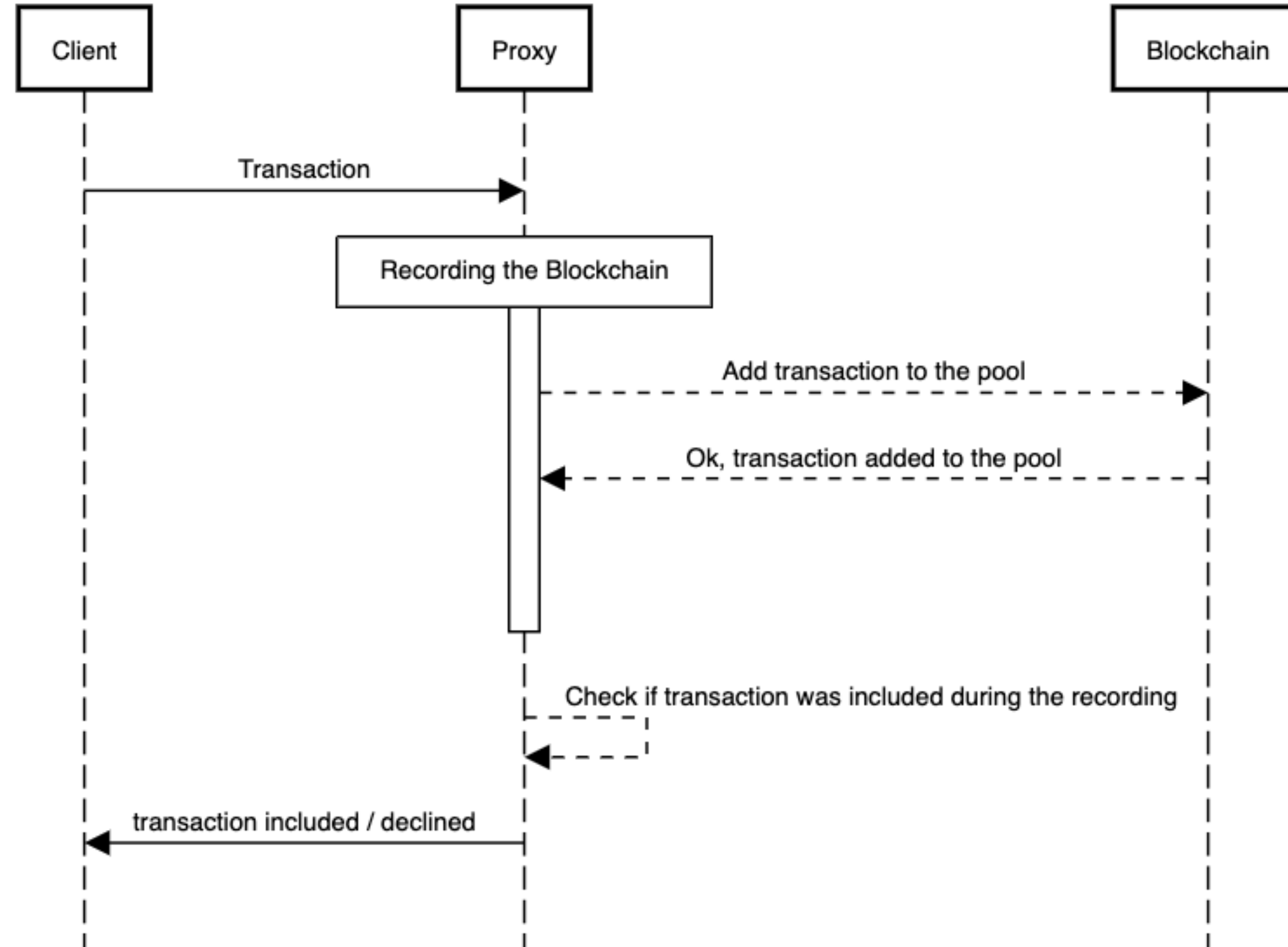
Production readiness

Performance

Observation on the load test

- The number of ballots the client perceived as being accepted did not match the number of ballots that were actually added to the blockchain.
- Investigation has to be done on the way transactions were handled

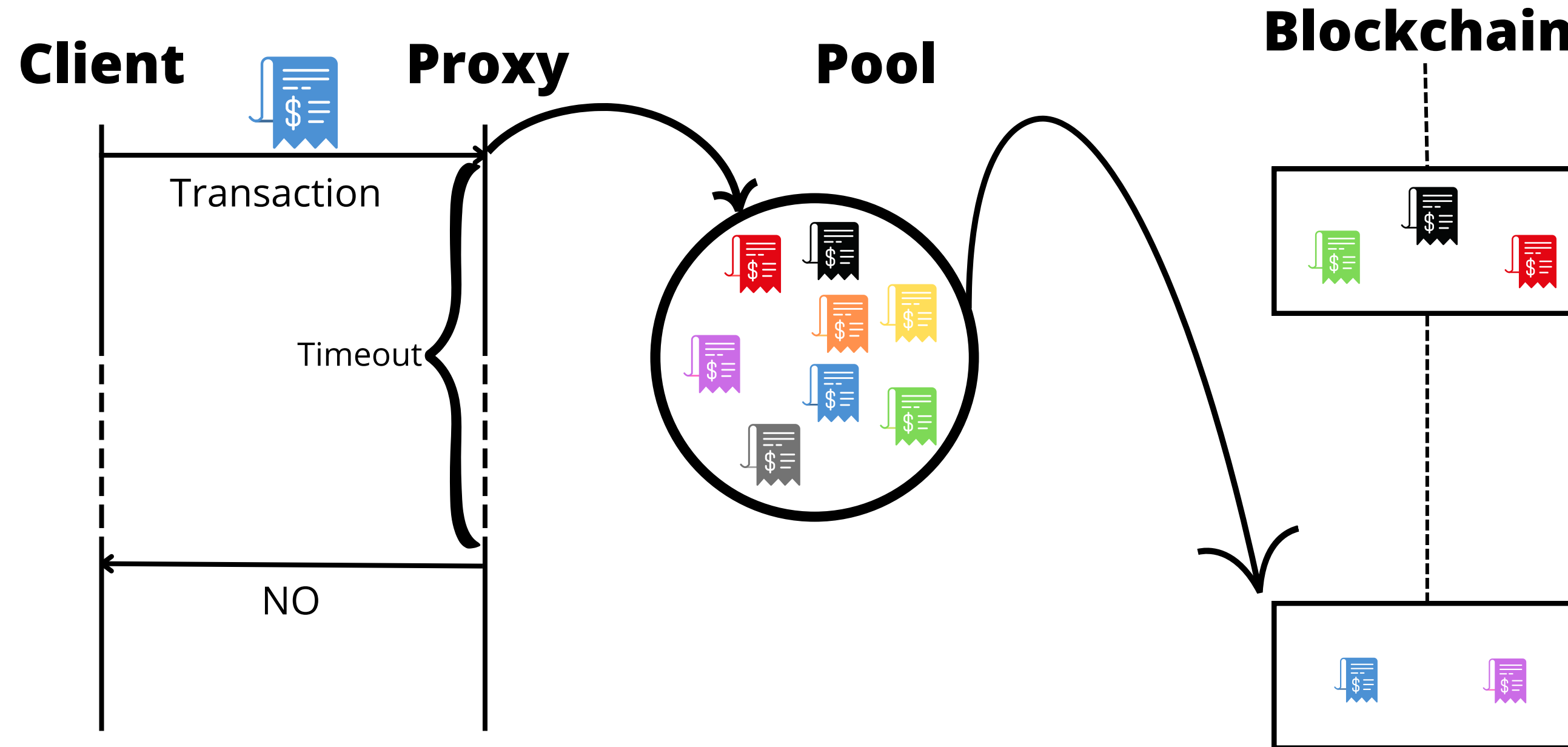
Previous Transaction System



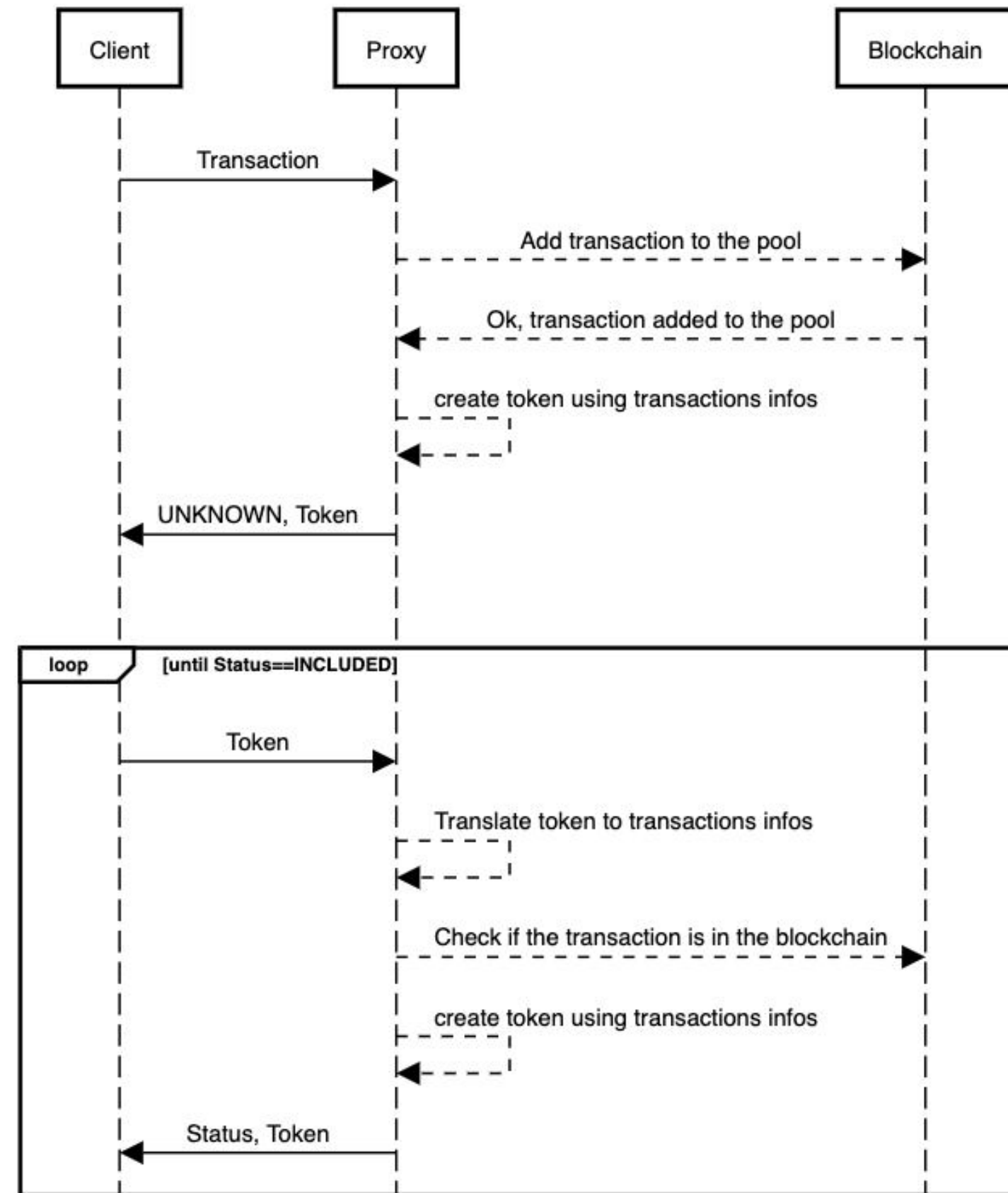
Production readiness

Transaction mechanism

Issue

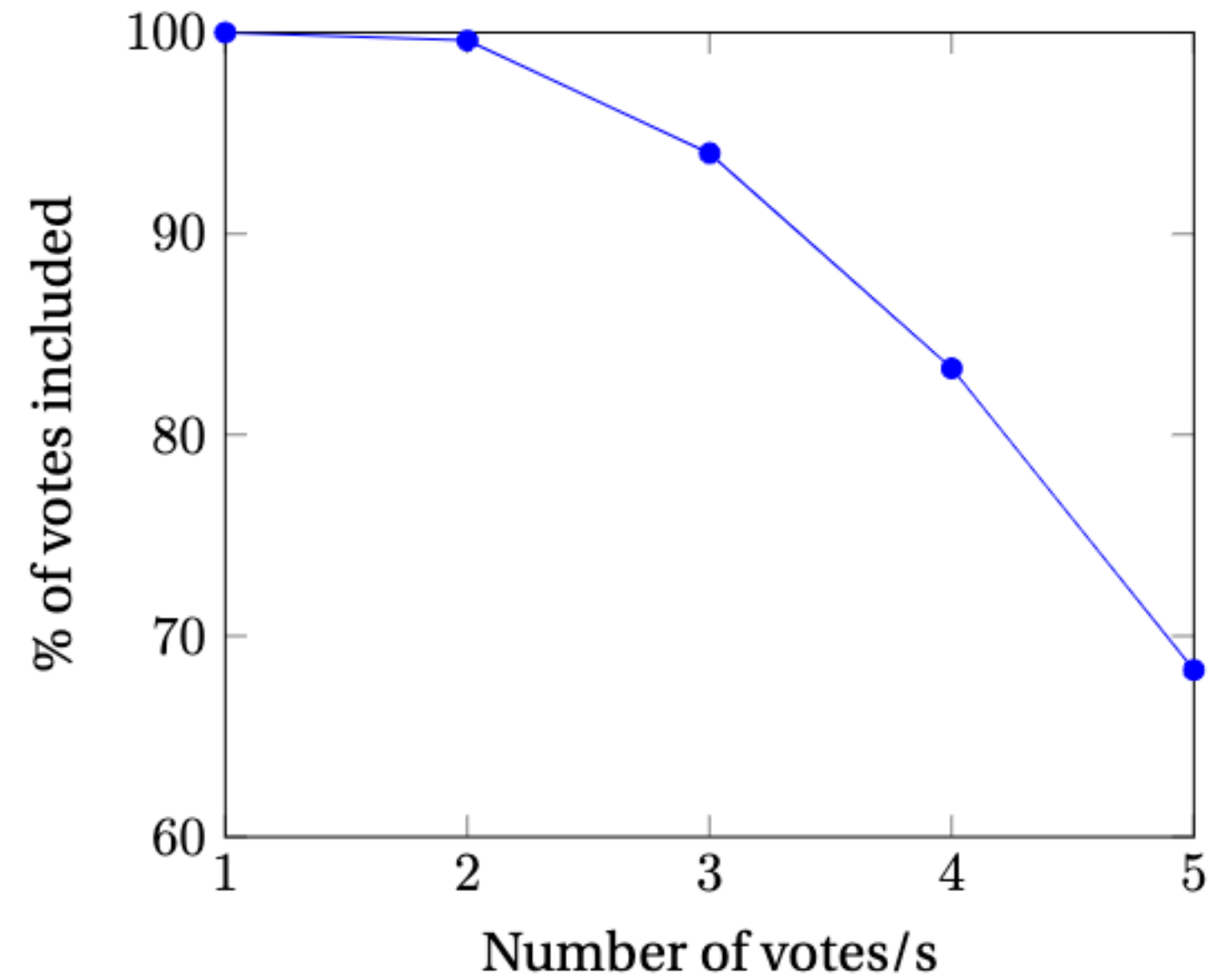


New Transaction System



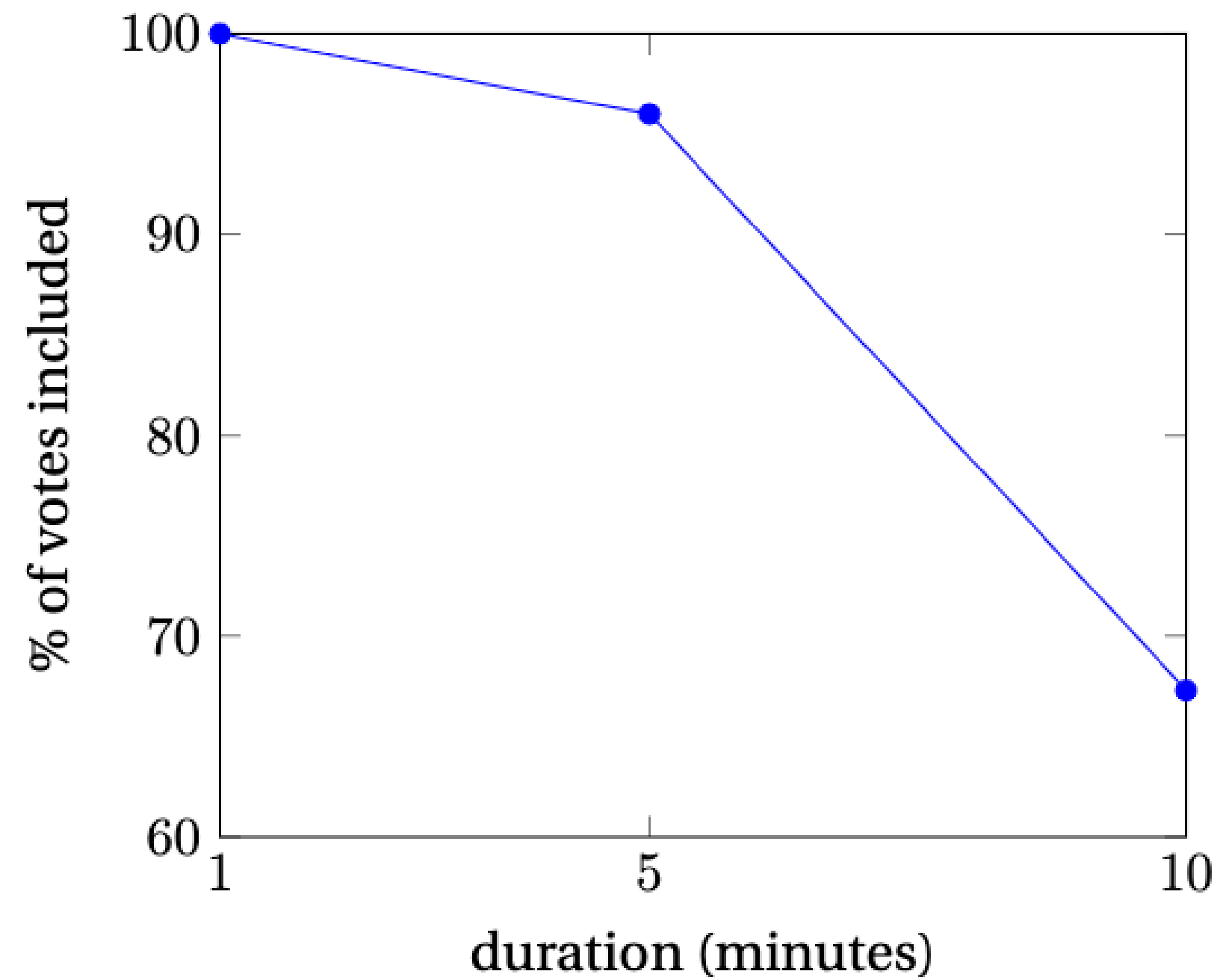
Analysis

% of votes included as a function of the number of votes per second for 60 second



Analysis

% of votes included with 1 vote/s with various duration



Production readiness

Performance

Conclusion of the load test

- Not all ballots sent are casted
- number of transactions in the pool \leq number of nodes

Production readiness

Future work

- More investigation on the issue

Production readiness

Future work

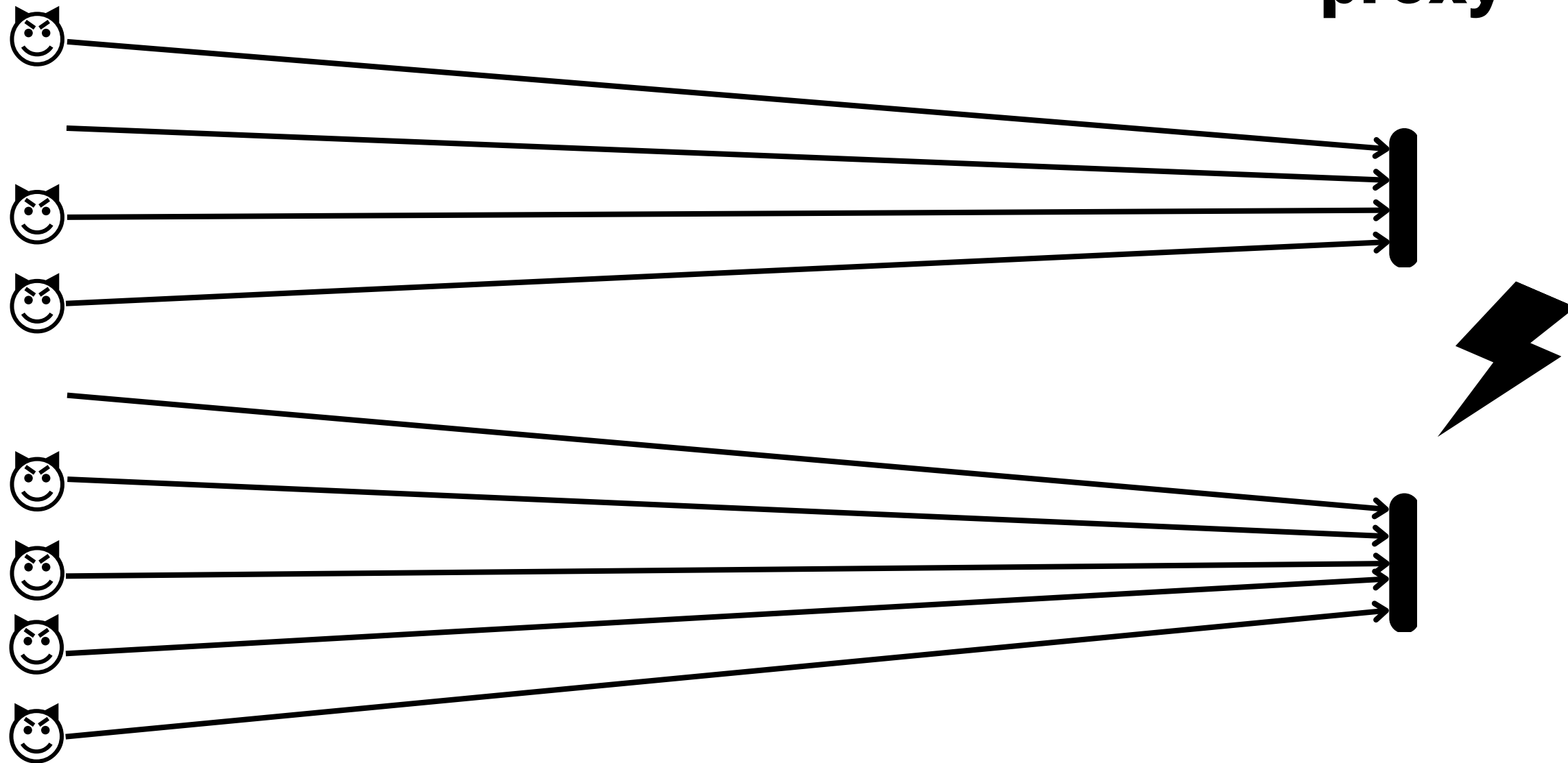
- More investigation on the issue
- Proxy splitting to be resilient to DOS

Proxy splitting

Before

Clients

proxy



Production readiness

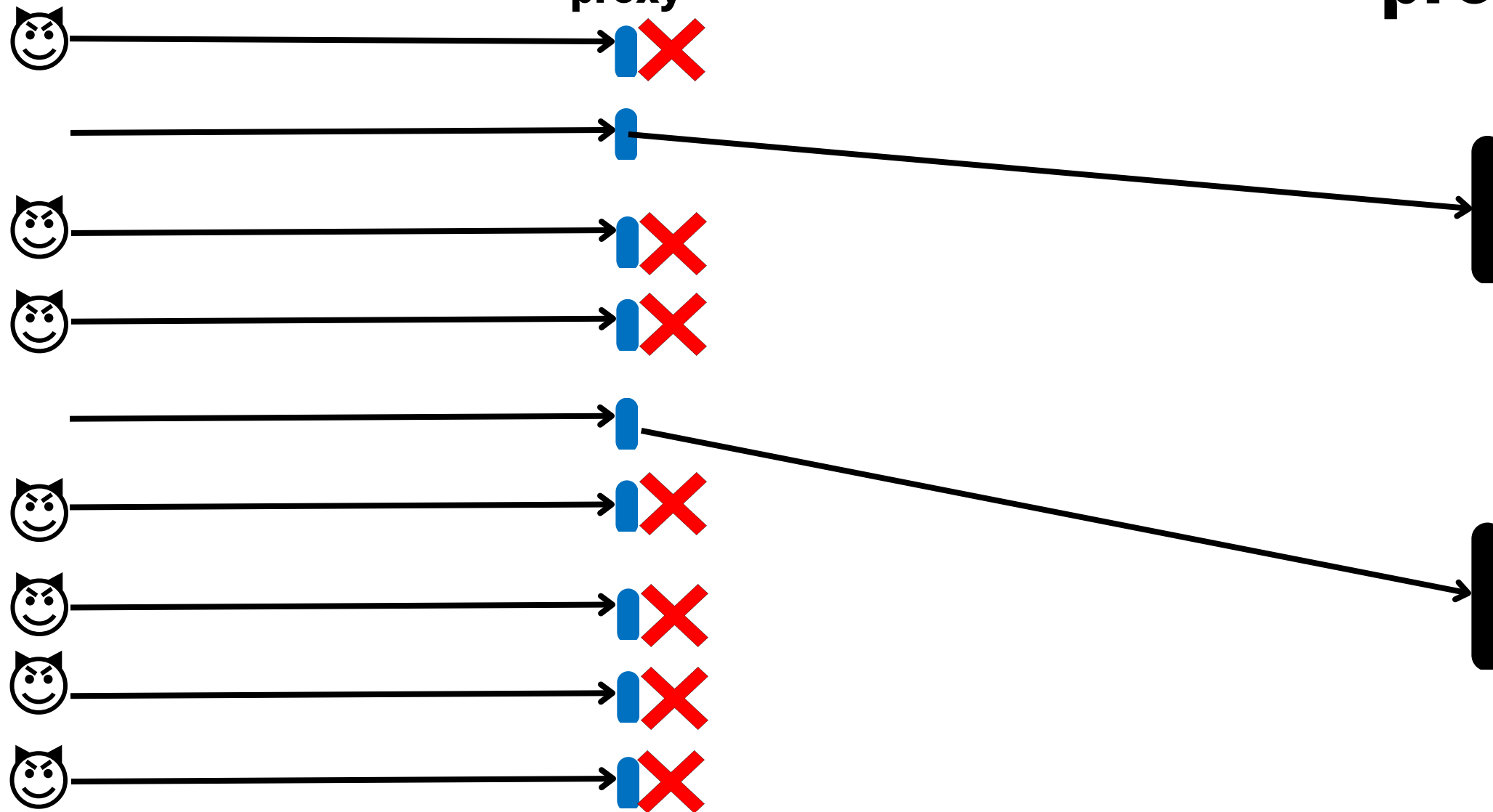
Proxy splitting

Clients

Cheap authentication

proxy

proxys

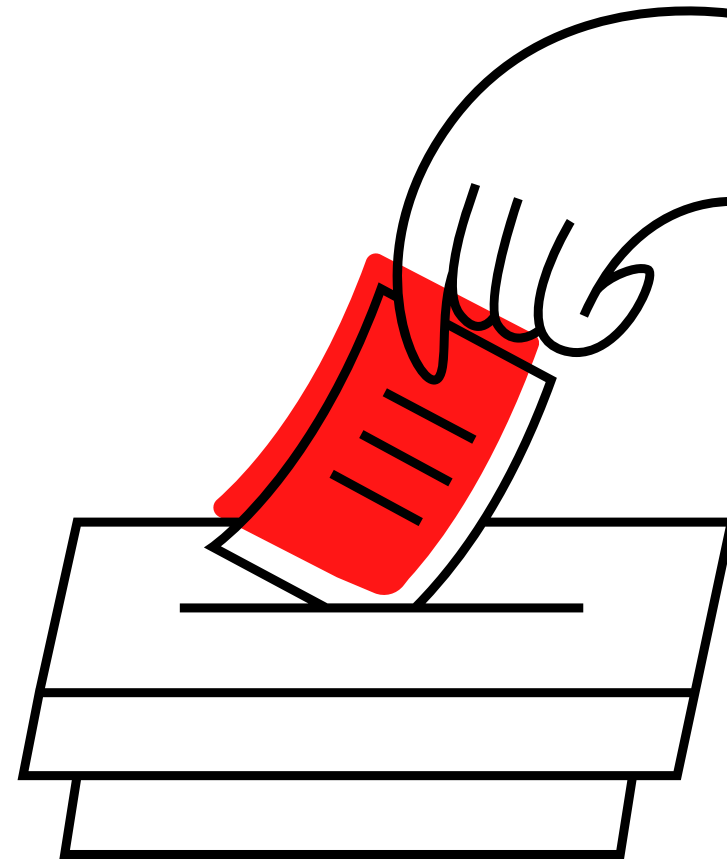


Production readiness

Future work

- More investigation on the issue
- Proxy splitting to be resilient to DOS
- Batch inclusion verification requests

Security audit



Chen Chang Lew (Zac)

Achieving Security Properties



Authorization



Confidentiality



Privacy



Integrity



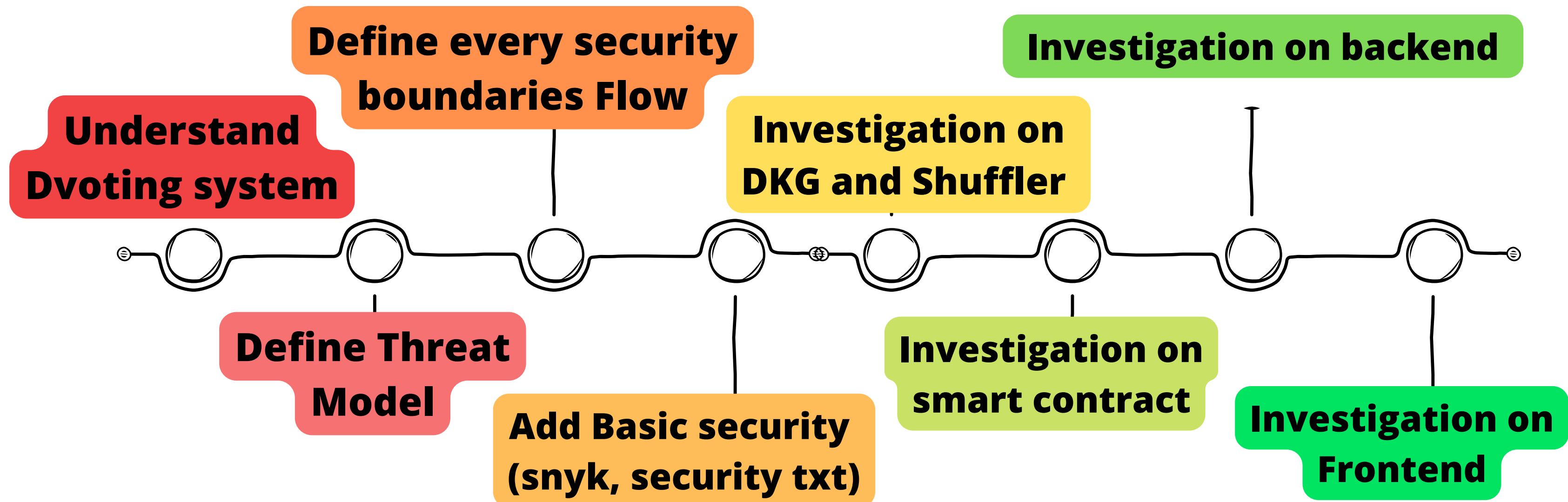
Verifiability



Availability

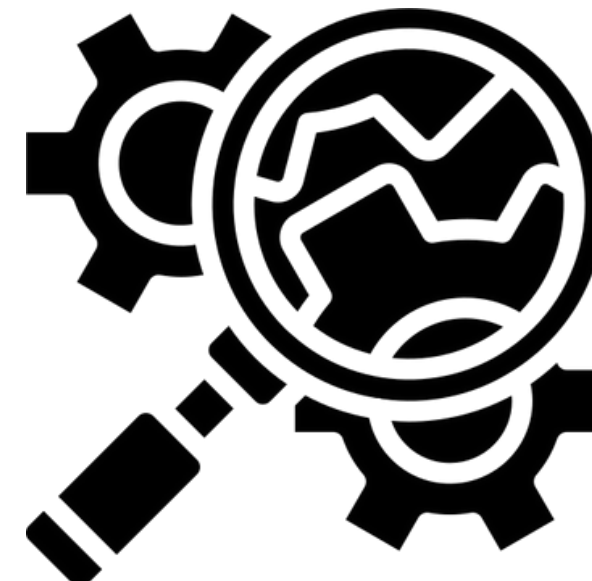
Improve product security

Plan for the semester



Summarize

- Create a **Threat Model** for D-voting
- Apply security scanning tools **SNYK** (scan **third-party** libraries)
- Add **security txt** (for researchers to report vulnerabilities)
- **Inspect** the whole codebase (found **9** security issues and **11** tech debt)
- Adding **vote verifiability** design and planning



Threat 5

Didn't check for
the maximum
length of the form.

≡ Text

Main properties

Title

test long text

Hint

long text should have a length

Answers

dos attack

+

Additional properties

Max number of choices

1

Min number of choices

0

MaxLength

888888888888

Regex

Enter your regex

Cancel

✓ Save

Threat 6

`contracts/evoting/evoting.go combineShares()`

Election not able
to review if anyone
submit a fake vote

```
for j := 0; j < ballotSize; j++ {  
    chunk, err := decrypt(i, j, allPubShares, form.PubsharesUnits.Indexes)  
    if err != nil {  
        return xerrors.Errorf("failed to decrypt (K, C): %v", err)  
    }  
    ...  
}
```

Threat 7

ShuffleThreshold
!=
nbrSubmissions
Threshold

ShuffleThreshold > f
nbrSubmissionsThreshold > 2f
f = # of malicious node

Login

Get Started



Threat 9

web/backend/src/Server.ts

















Only Admin and operator can vote

```
// Secure /api/evoting to admins and operators
app.use('/api/evoting/*', (req, res, next) => {
  if (!isAuthorized(req.session.userid, SUBJECT_ELECTION, ACTION_CREATE)) {
    res.status(400).send('Unauthorized - only admins and operators allowed');
    return;
  }
  next();
});
```

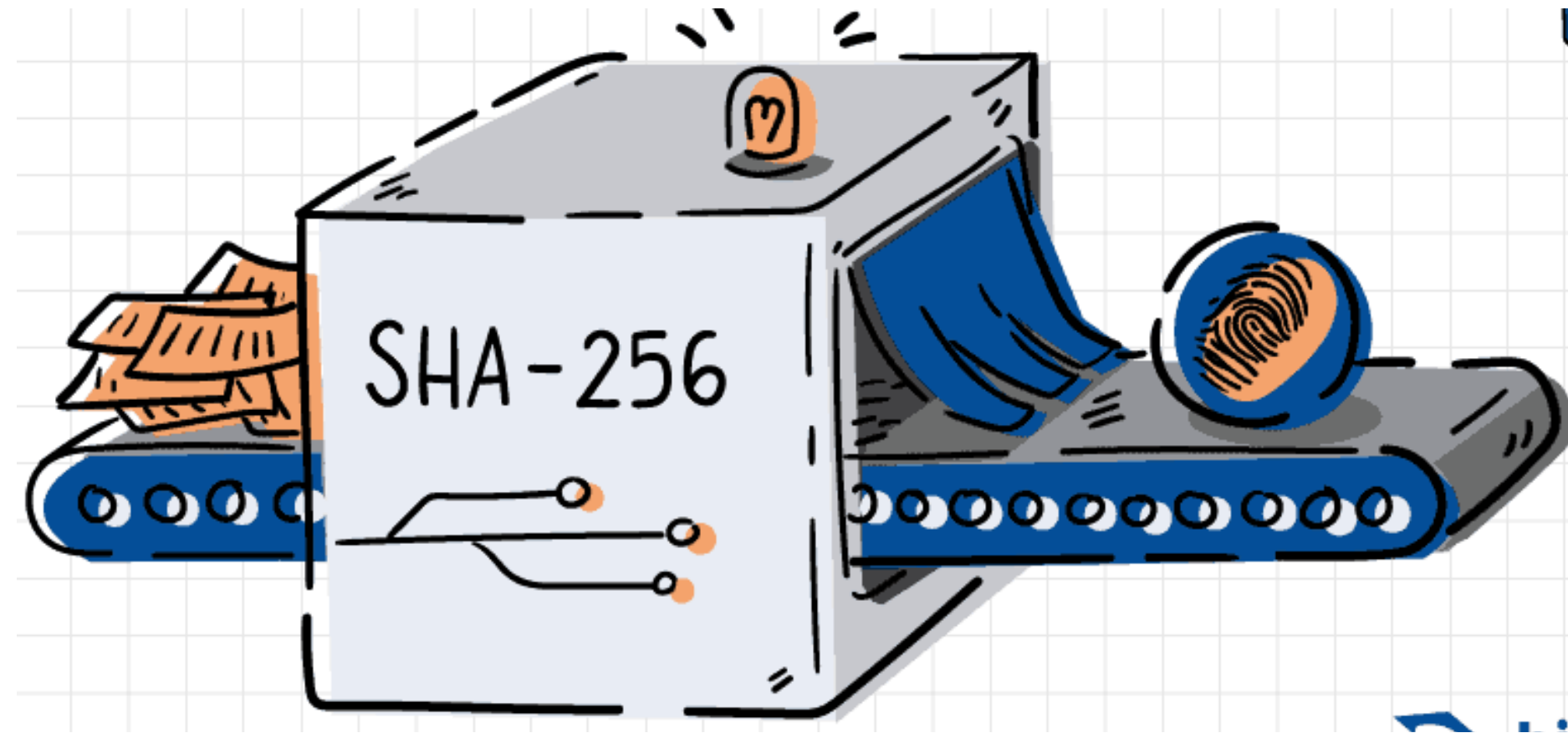
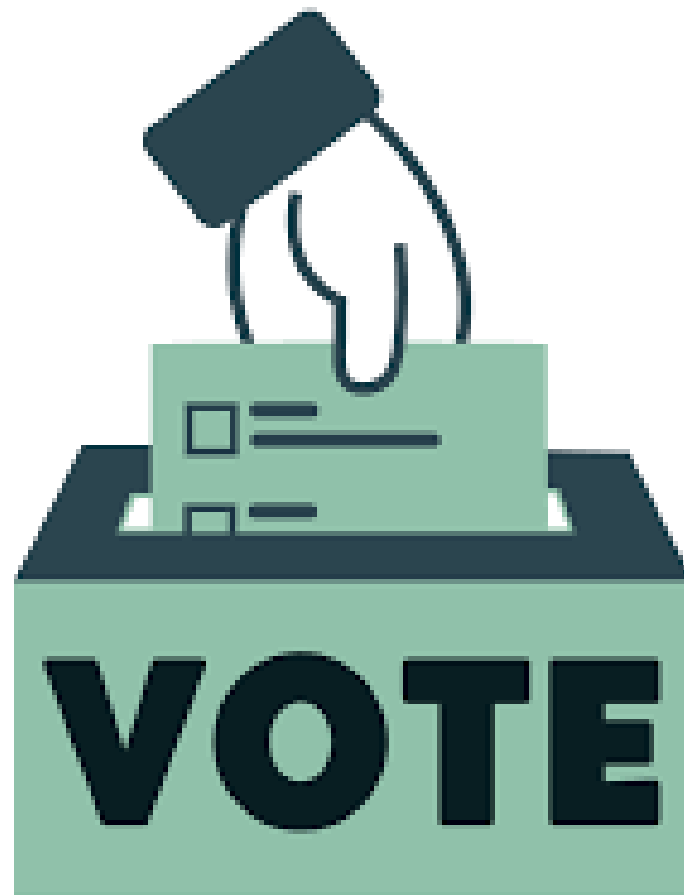
Created Issue (Security)

<input type="checkbox"/>	<input checked="" type="radio"/> THREAT - A user who is not an admin or operator cannot vote. security issue web backend #253 opened last month by Flamewind97
<input type="checkbox"/>	<input checked="" type="radio"/> THREAT - Logout will not clear all the sessions in the browser security issue web frontend #252 opened last month by Flamewind97
<input type="checkbox"/>	<input checked="" type="radio"/> THREAT - ShuffleThreshold shouldn't be used as the nbrSubmissions threshold security issue smart contract #251 opened last month by Flamewind97
<input type="checkbox"/>	<input checked="" type="radio"/> THREAT - Election will not be able to reveal the result if anyone submits a fake vote. security issue smart contract #250 opened last month by Flamewind97
<input type="checkbox"/>	<input checked="" type="radio"/> THREAT - Frontend create form didn't check for the maximum length of the form security issue smart contract web frontend #249 opened last month by Flamewind97
<input type="checkbox"/>	<input checked="" type="radio"/> THREAT - A user can vote multiple times (count as multiple votes) in an election. security issue web backend #248 opened last month by Flamewind97
<input type="checkbox"/>	<input checked="" type="radio"/> THREAT - All the election stages can be ignored by malicious node security issue web backend #247 opened last month by Flamewind97
<input type="checkbox"/>	<input checked="" type="radio"/> THREAT - The public/private key of the election is changed by the Adversary security issue web frontend #246 opened last month by Flamewind97
<input type="checkbox"/>	<input checked="" type="radio"/> THREAT - Denial of Service, Dkg public key will always return false if an adversary compromise one device. security issue dkg #217 opened on Nov 23, 2022 by Flamewind97

Created Issue (Tech Debt)

<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - Remove point & public key related in web backend since we didn't use it. web backend</div></div> <div>#245 opened last month by Flamewind97</div>		
<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - shouldn't use fingerprint function for pseudorandomness because it is not efficient. smart contract shuffling</div></div> <div>#244 opened last month by Flamewind97</div>		
<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - change loop and sleep to channel + ctx timeout to increase readability of code. shuffling</div></div> <div>#221 opened on Nov 23, 2022 by Flamewind97</div>		
<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - Need refactor in dkg & shuffler dkg shuffling</div></div> <div>#220 by Flamewind97 was closed last month</div>	 1	
<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - Duplicate function getForm() dkg shuffling</div></div> <div>#219 opened on Nov 23, 2022 by Flamewind97</div>	 1	
<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - Encrypt function in DKG & Marshall ballot function in smart contract is not used anymore dkg smart contract</div></div> <div>#218 opened on Nov 23, 2022 by Flamewind97</div>	 1	
<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - check lenAddrs before sending getPeerKey dkg</div></div> <div>#216 by Flamewind97 was closed on Dec 15, 2022</div>	 1	
<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - variable name "buff, formIDBuf, formIDBuff, formID" not consistent proxy smart contract</div></div> <div>#214 opened on Nov 16, 2022 by Flamewind97</div>	 1	
<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - unclear/wrong comment proxy shuffling</div></div> <div>#213 opened on Nov 16, 2022 by Flamewind97</div>	 1	
<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - change legacy structure "CreateForm" smart contract</div></div> <div>#212 opened on Nov 16, 2022 by Flamewind97</div>		
<input type="checkbox"/>	<div><div>🕒</div><div>Technical Debt - verify signature before execute request proxy</div></div> <div>#210 by Flamewind97 was closed on Dec 15, 2022</div>	 1	

Verifiability



Verifiability

Election details	
User xxx	e46a22f8d4c8 855603b27e0 cdb22ae4118d
User xxx	65f6acbd45ef db13174f27ea 35c427e531fe

Future Work

- Conduct a review of **Dela and Kyber** crypto package to identify any additional security risks because we assume them to be safe in the security report.
- Review and add more **security measure tools** for d-voting to ensure that it meets current security best practices.
- Conduct **regular security assessments** to identify and address any new or emerging security threats.
- Patch the **security issues** and technical debts.

Thanks for your time !

