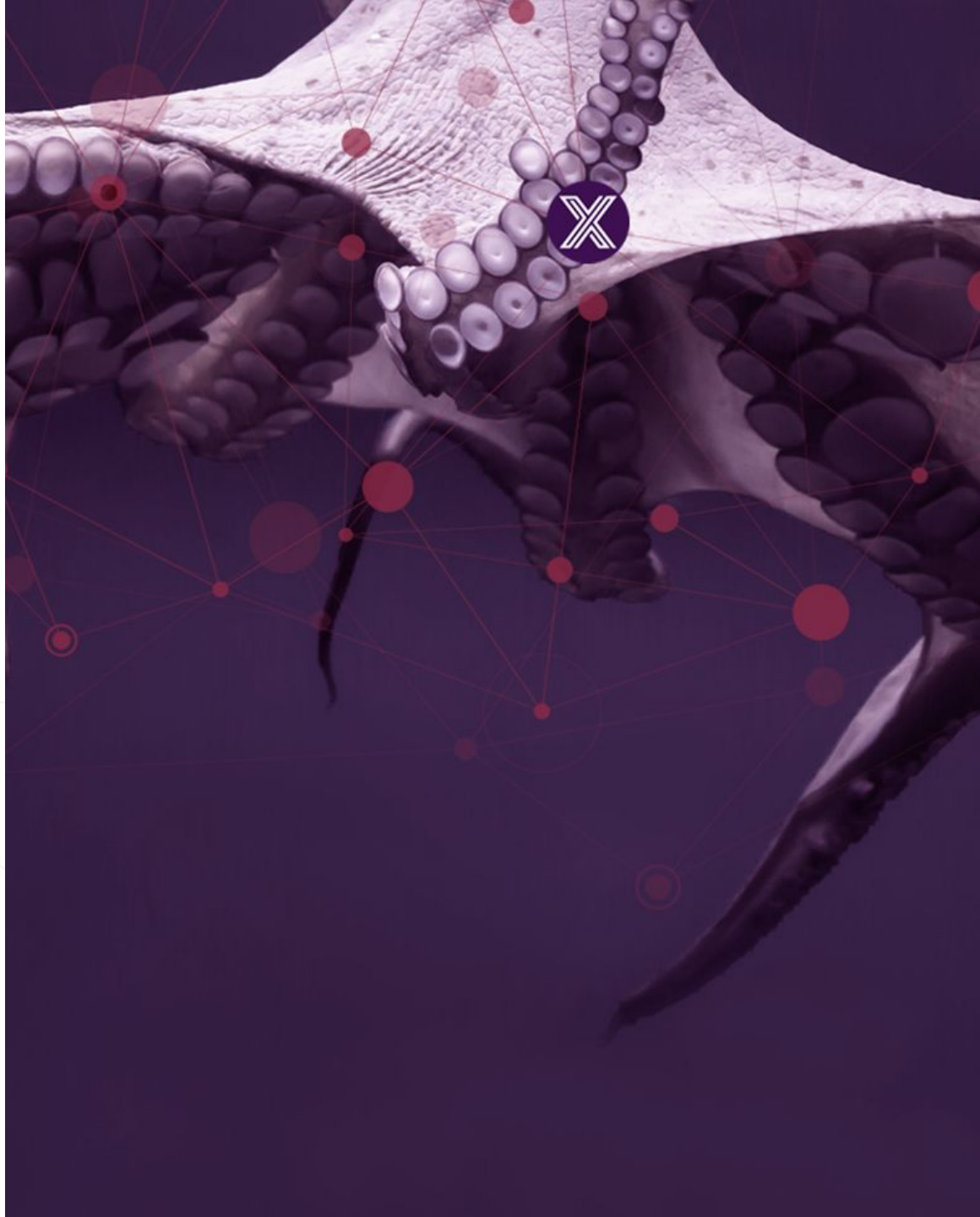
EDGE X FOUNDRY™

EdgeX Auth Service

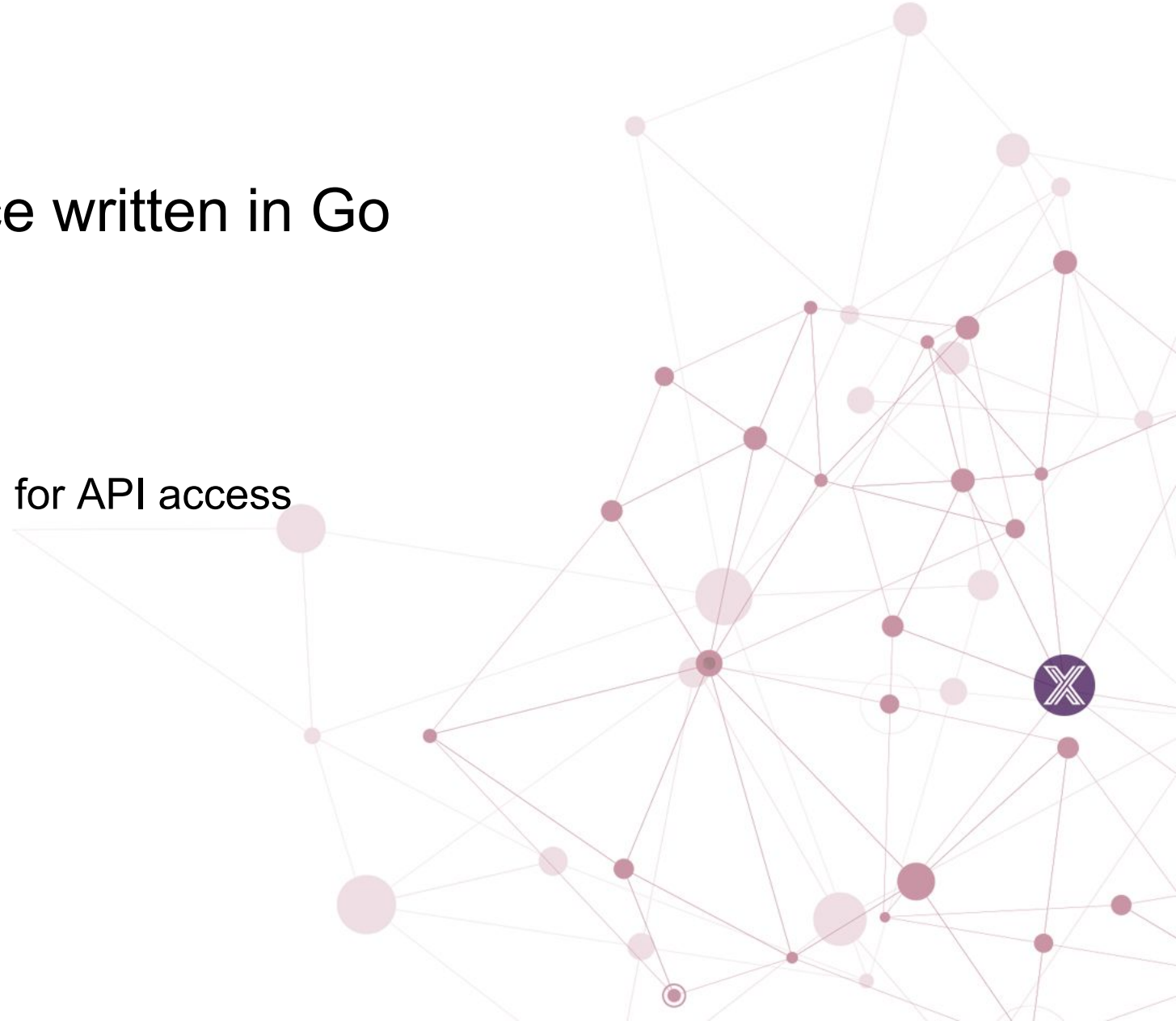
Drasko Draskovic – Mainflux

drasko@mainflux.com



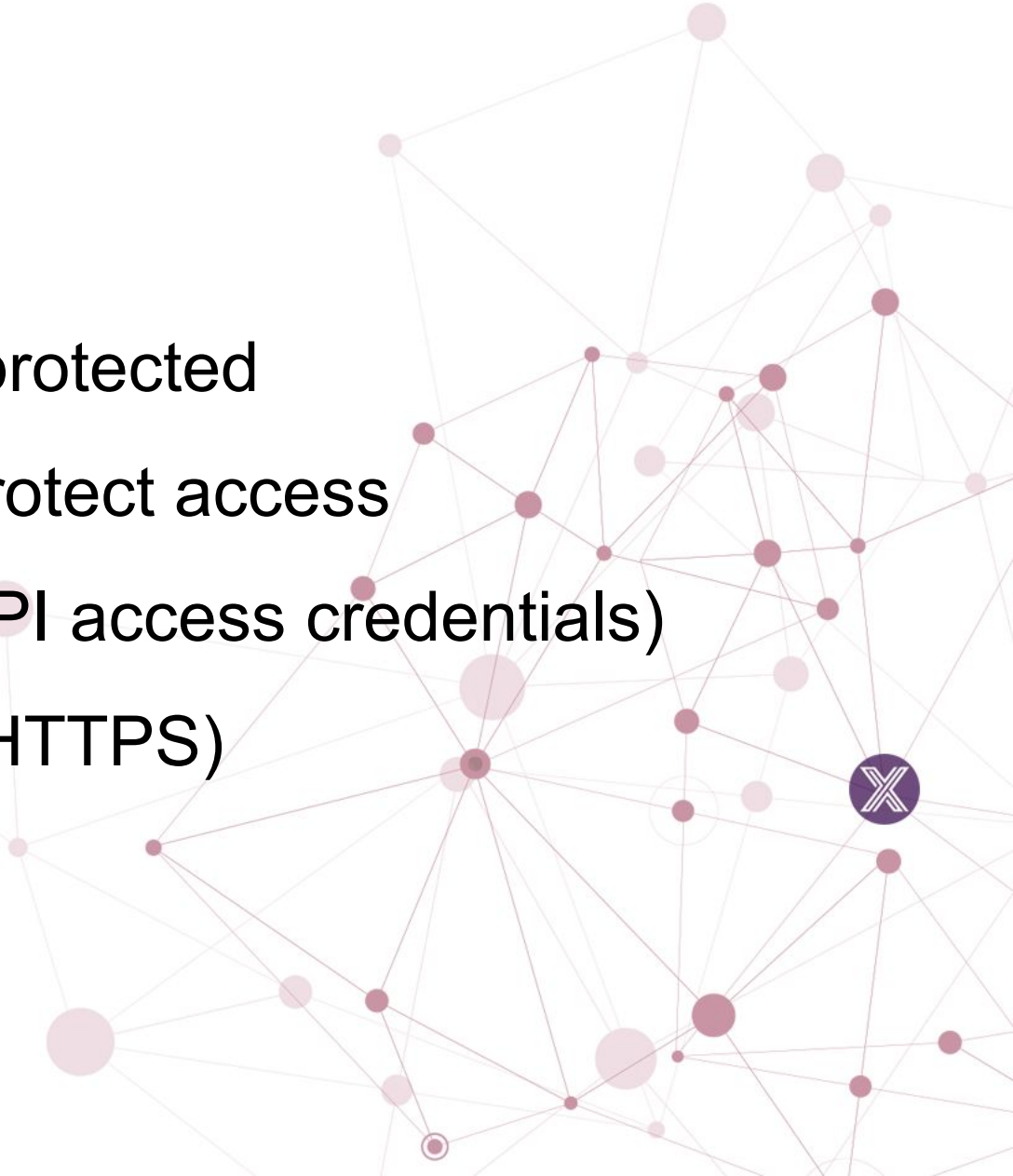
Abstract

- EdgeX AuthX/AuthZ service written in Go
- Enables:
  - User management (CRUD)
  - User login – produces JWT token for API access
  - TLS Encryption (HTTPS)



Problem

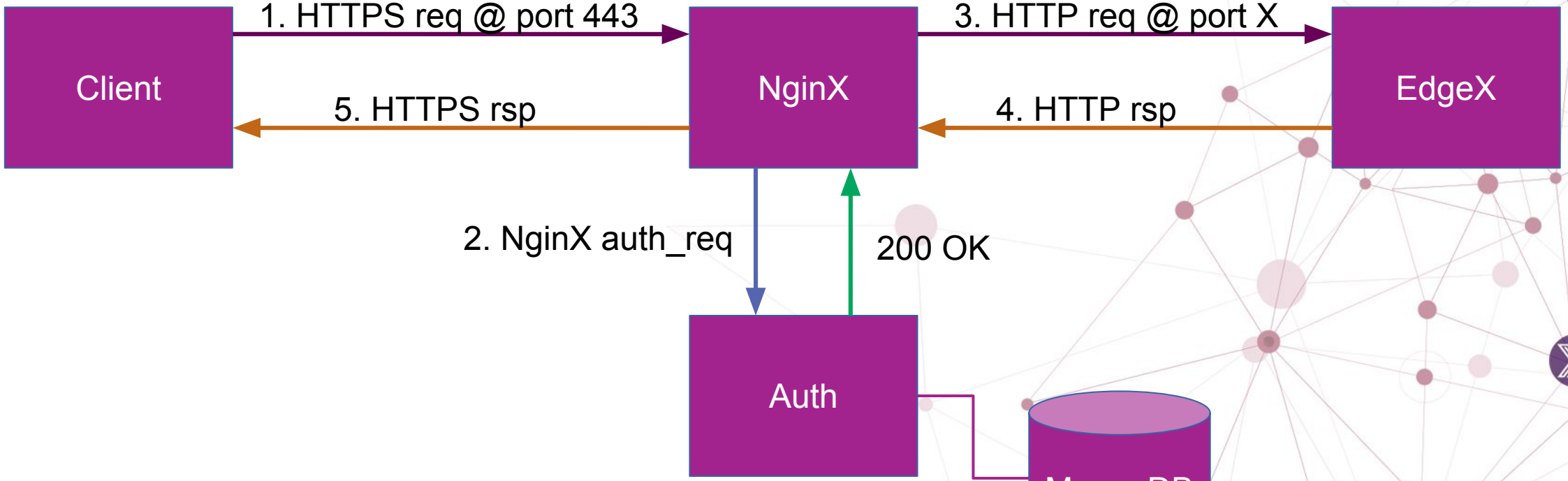
- EdgeX API lacks basic security
- Microservices API access should be protected
- Reverse proxy (NginX, Traefik) can protect access
- User entities should exist (to obtain API access credentials)
- Traffic should be encrypted via TLS (HTTPS)



Solution

- Auth microservice that sits behind reverse proxy (NginX or Traefik)
- Leverage on NginX/Traefik standard “auth_req” forwarding feature
- Every request is forwarded automatically by reverse proxy to Auth microservice that checks credentials and grants access
- If everything is OK – reverse proxy let the req pass
- TLS encryption can be terminated on reverse proxy – they are good at exposing HTTPS endpoints and checking certificates, then proxying to plain HTTP within the system

Diagram


Client
1. HTTPS req @ port 443
5. HTTPS rsp
NginX
3. HTTP req @ port X
4. HTTP rsp
EdgeX
2. NginX auth_req
200 OK
Auth
MongoDB

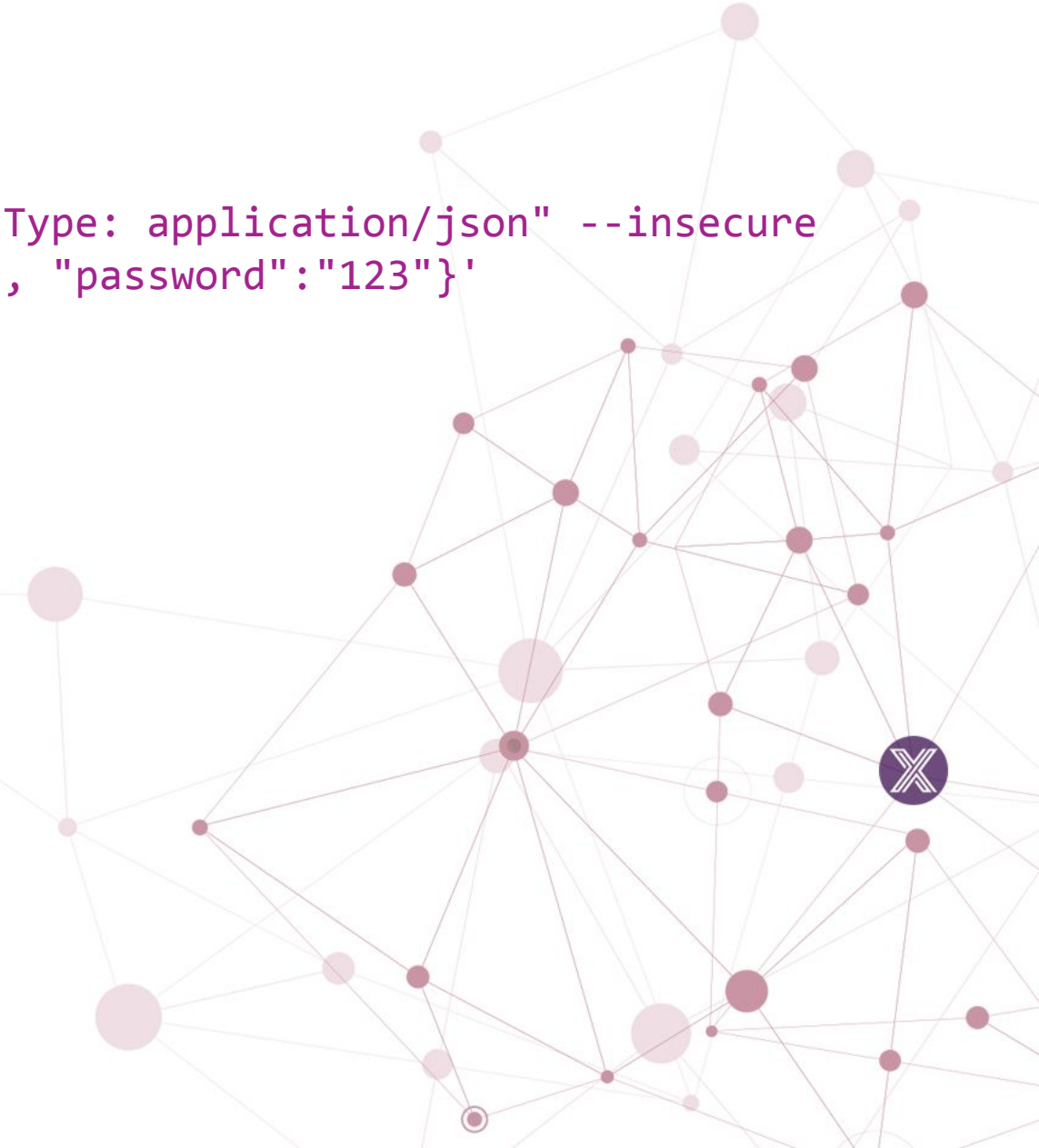
Reverse Proxy Request Forwarding

- NiginX auth_request: http://nginx.org/en/docs/http/ngx_http_auth_request_module.html
- Treafik Forward Authentication: https://docs.traefik.io/configuration/entrypoints/#forward-authentication
- Consul-template: https://medium.com/@ladislavGazo/easy-routing-and-service-discovery-with-docker-consul-and-nginx-acfd48e1a291 and https://github.com/hashicorp/consul-template/blob/master/examples/nginx.md
- Hydra Ladon: https://github.com/ory/ladon

User Creation

```
drasko@Marx:~$ curl -s -S -i -X POST -H "Content-Type: application/json" --insecure
https://localhost/users -d '{"username":"jon.doe", "password":"123"}'
```

```
HTTP/2 201
server: nginx/1.13.9
date: Tue, 20 Mar 2018 21:39:38 GMT
content-type: text/plain; charset=utf-8
content-length: 0
location: /users/4422b1e5-0489-4a45-8562-107a2c80e9d2
strict-transport-security: max-age=63072000; includeSubdomains
x-frame-options: DENY
x-content-type-options: nosniff
access-control-allow-origin: *
access-control-allow-methods: *
access-control-allow-headers: *
```



User Login (Obtain JWT)

```
drasko@Marx:~$ curl -s -S -i -X POST -H "Content-Type: application/json" --insecure
https://localhost/login -d '{"username":"jon.doe", "password":"123"}'

HTTP/2 201
server: nginx/1.13.9
date: Tue, 20 Mar 2018 21:49:12 GMT
content-type: text/plain; charset=utf-8
content-length: 161
strict-transport-security: max-age=63072000; includeSubdomains
x-frame-options: DENY
x-content-type-options: nosniff
access-control-allow-origin: *
access-control-allow-methods: *
access-control-allow-headers: *

{"token":"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpYXQiOjE1MjE1ODI1NTIsImlzcyI6Im1haW5m
bHV4Iiwic3ViIjoiam9uLmRvZSJ9.oBl9Medafi-cAw0xTKtnyR8n2MTMTQ3Ilz5JjGIPnf0"}
```

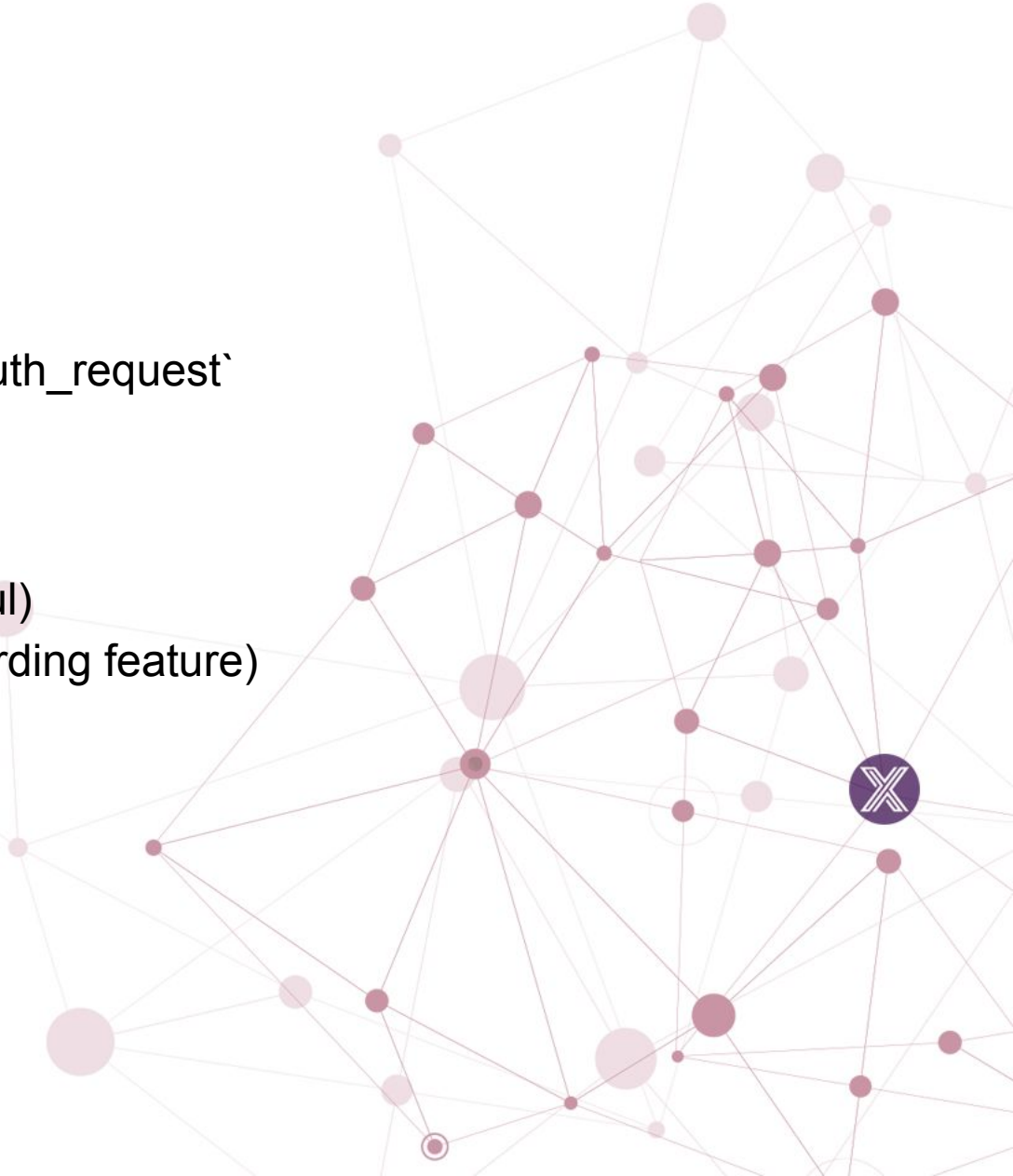
Sending Requests With JWT

- Token should be put into “Authorization” header:

```
drasko@Marx:~$ curl -s -S -i -X GET -H "Content-Type: application/json" -H
"Authorization:
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpYXQiOjE1MjE1ODI2ODgsImlzcyI6Im1haW5mbHV4Iiwic3
ViIjoiam9uLmRvZSJ9.vU2g5faJRXORD6ZYiZtB73NhOALlHvqgL0ew9Jh155c" --insecure
https://localhost/api/hello
```

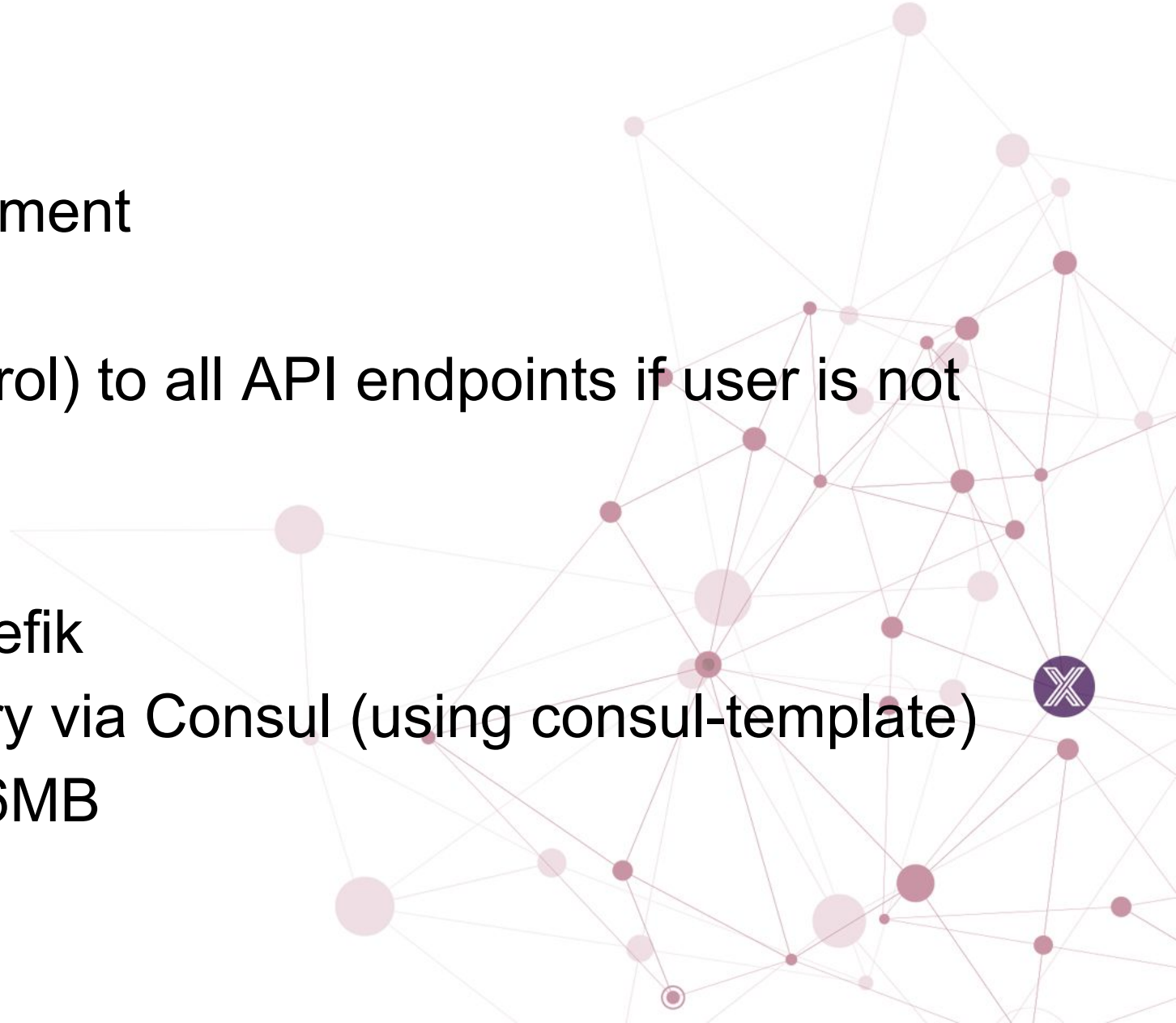
Status

- Currently:
  - HTTPS (TLS v1.2) is working
  - NginX is forwarding all requests to Auth service via `auth_request`
  - User management and login
- In progress:
  - Consul auto-discovery support (NginX can read Consul)
  - Traefik support (Traefik also has `auth_request` forwarding feature)
- Future:
  - Fine grained AuthZ via Hydra Ladon



Benefits

- User creation and management
- User login via JWT token
- Authorization (access control) to all API endpoints if user is not logged in
- TLS encryption (HTTPS)
- Support for NginX and Traefik
- Automatic service discovery via Consul (using consul-template)
- Small Docker image – 5.46MB



References

- GitHub Repo: https://github.com/drasko/edgex-auth
- E-mail: drasko@mainflux or janko@mainflux

Thank You

Mainflux Team