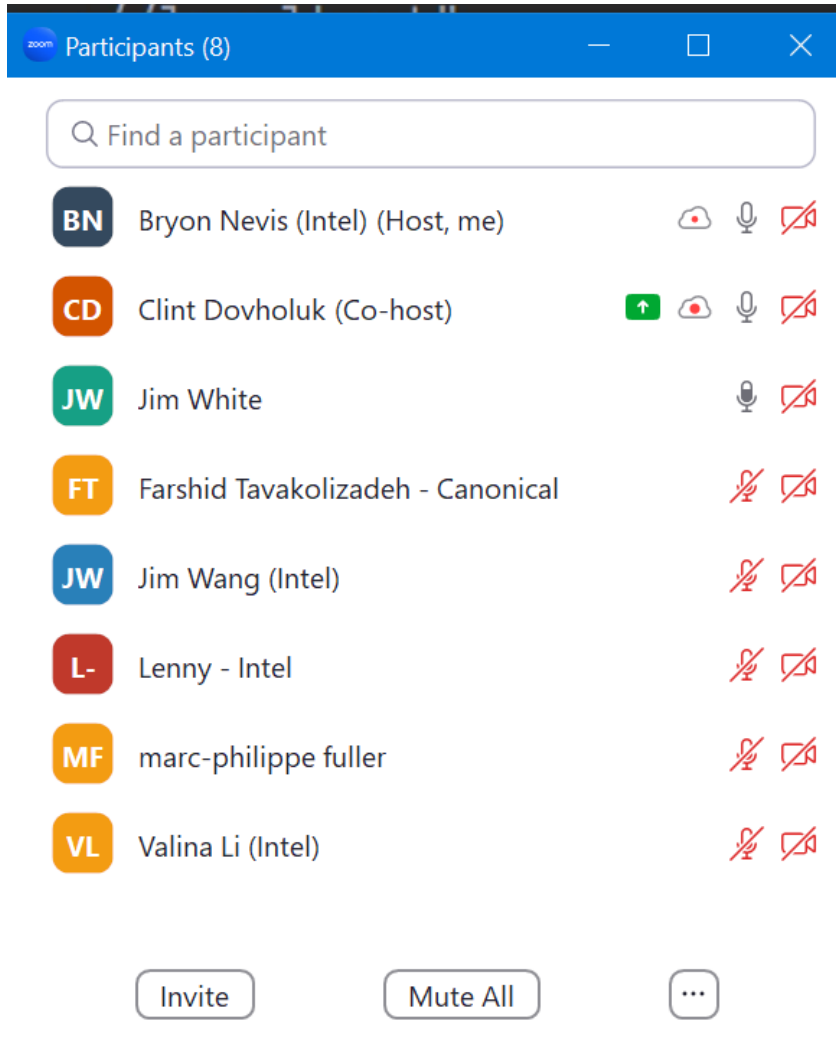


EdgeX Security WG Meeting

<https://wiki.edgexfoundry.org/display/FA/Security+Working+Group>

January 11, 2023

Attendees



The screenshot shows a Zoom meeting window titled "Participants (8)". It features a search bar at the top with the text "Find a participant". Below the search bar is a list of eight participants, each with a colored initials icon, their name, and their role. To the right of each name are icons for video, microphone, and chat. At the bottom of the list are three buttons: "Invite", "Mute All", and a three-dot menu icon.

Initials	Name	Role	Video	Audio	Chat
BN	Bryon Nevis (Intel)	(Host, me)	Off	Off	Off
CD	Clint Dovholuk (Co-host)	(Co-host)	On	Off	Off
JW	Jim White		Off	Off	Off
FT	Farshid Tavakolizadeh - Canonical		Off	Off	Off
JW	Jim Wang (Intel)		Off	Off	Off
L-	Lenny - Intel		Off	Off	Off
MF	marc-philippe fuller		Off	Off	Off
VL	Valina Li (Intel)		Off	Off	Off

Agenda

TSC Update

- STRIDE Threat Model hopefully merge today or in the next day or two

- Fix for GHSA-xrjj-mj9h-534m and GHSA-69ch-w2m2-3vjp bubbling through with go-mod-secrets update
- Microservice authentication UCR voting underway (docs#920). ADR updates next.
- OpenZiti and Vault JWT authentication prototypes are well along
 - JWT prototype: 80MB RAM saved (160->80), 329MB images saved (docker, 353->24)
 - OpenZiti: no numbers yet (docker stats, docker images) before and after

Opens

- Clint to demo OpenZiti
 - Have added OpenZiti to go-mod-bootstrap

```

for serviceKey, serviceInfo := range config.GetBootstrap().Clients {
    var url string
    var err error

    if len(serviceInfo.ZeroTrustOptions) > 0 {
        if idFile, ok := serviceInfo.ZeroTrustOptions["IdentityFile"]; ok {
            _, err := ziti.LoadContext(idFile)
            if err != nil {
                panic(err)
            }
        }
    }
}

```

```

[Clients]
  [Clients.core-metadata]
    Protocol = "http"
    Host = "localhost"
    Port = 59881
  [Clients.core-metadata.ZeroTrustOptions]
    IdentityFile="c:/temp/svc.core-command.identity.json"

```

-
- With this change, outgoing requests will go over a zero trust transport.
- For listeners:

```

CURSExposeHeaders = "Cache-Control, Content-Language, Content-Length, Content-Type"
CORSMAXAge = 3600
[Service.ListenOptions]
  IdentityFile="c:/temp/svc.core-command.identity.json"
  ServiceName="svc.core-command"

```

-

```

switch bootstrapConfig.Service.ListenMode {
case "zerotrust":
    lc.Info(msg: "using zerotrust - look at you go")

    idfile := bootstrapConfig.Service.ListenOptions["IdentityFile"]
    identityConfig, err := config.NewFromFile(idfile)

    if err != nil {
        lc.Errorf(msg: "could not load configuration file: %v", err)
    }

    ctx := ziti.NewContextWithConfig(identityConfig)

    if err = ctx.Authenticate(); err != nil {
        lc.Errorf(msg: "could not authenticate: %v", err)
    }

    serviceName := bootstrapConfig.Service.ListenOptions["ServiceName"]
    ln, err = ctx.Listen(serviceName)

    if err != nil {
        log.Fatalf(format: "could not bind service %s: %v", serviceName, err)
    }

    /*
    if args.TlsConfig != nil {
        ln = tls.NewListener(ln, args.TlsConfig)
    }
    */

case "http":
default:
    lc.Warn(msg: "using ListenMode 'http'")
    ln, err = net.Listen(network: "tcp", addr)
}

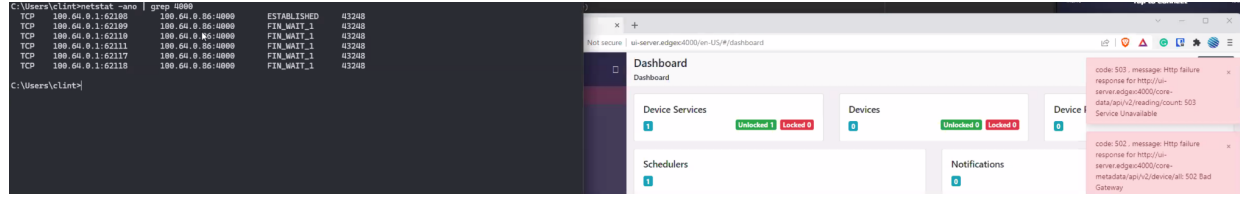
```

-
- Connecting to UI over zero trust network:

Sort By Name ▾			Name	_cdaws_clint
	Clint-Posture	1	On	
	NF_Clint_RamcoDev	23	On	
	_cdaws_clint	1	On	
	_dovanet_ClintRemoteWind	10	On	
	clint_dovholuk-bastion1	9	On	
	mattermost.clint.dovholuk.	2	On	
	vault.client	1	On	

Filter			Sort By Name ▾
	svc.ui-server	ui-server.edgex	4000

-
- Note: despite connecting to port 4000, there is no listening port on 4000:



-
-

- Discussion to follow
 - Limitation: Can't Zitify non-edgeX services like Redis, Mosquitto, etc easily without using LD_PRELOAD.
 - Leaves a hole for cross-network communication
 - Could send the external services via a tunneler
 - Using Vault to control access to zero trust overlay
 - Need to distribute PKI
 - Overlay will require controller and a router
 - Clint's perspective: would naturally assume an OpenZiti network already exists
 - THINK ABOUT
 - What if EdgeX provided multiple compose files that needed to be run to bring up an EdgeX system
 - Base services: redis, mosquitto, consul, etc
 - Zero trust: openziti router and controller
 - EdgeX services themselves: data, metadata, command, ...
 - Current limitation: Openziti must always read key and certificate off of disk. If get key/cert from vault, need to buffer it on disk.
 -

Standing Agenda

- [Review Security Board](#)
- [Review CIS docker scan](#) (will skip unless something changes) (click latest run, go to classic, view console output).
 - Last checked: August 3, 2022 – as expected
- [Review Snyk \(Jenkins\)](#) (will skip unless something changes) ([Imagelist](#))
- Review action items from previous week

Action Items

- 7/14/21: Bryon: Update security policy documentation w.r.t. when to use GitHub security advisories to notify users of issues.