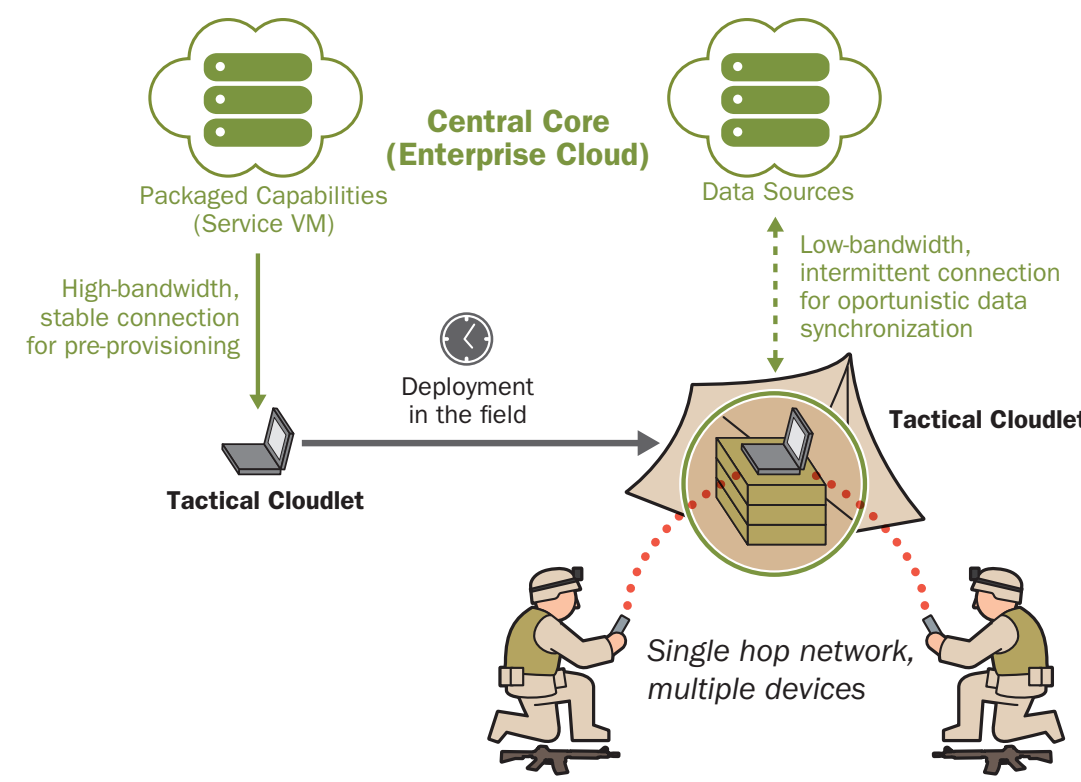# Tactical Computing and Communications (TCC)
Secure and Efficient Computing and Communications at the Edge

Previous Work
Tactical Cloudlets

Central Core (Enterprise Cloud)

Packaged Capabilities (Service VM)

Data Sources

High-bandwidth, stable connection for pre-provisioning

Low-bandwidth, intermittent connection for oportunistic data synchronization

Deployment in the field

Tactical Cloudlet

Tactical Cloudlet

Single hop network, multiple devices

Forward-deployed, discoverable, virtual machine (VM) based cloudlets that can be hosted on vehicles or other platforms
· computation offload
· forward data-staging
· filtering of data intended for mobile devices
· collection points for data heading for enterprise repositories

Features:
· Pre-Provisioned Cloudlets w/ App Store
· Standard Packaging of Service VMs
· Optimal Cloudlet Selection
· Cloudlet Management Console
· Cloudlet Handoff/ Migration
· Secure Key Generation and Exchange

Delay-Tolerant Networking (DTN)

	← Connected →	← Disconnected →	← Reconnecting →
GOALS	Maintain shared group context Make best use of available bandwidth	Applications continue to function Predict state where possible	Re-establish shared group context as quickly and accurately as possible
DTN NODE TASKS	Pre-cache data likely to be relevant later in the mission Delay transmission of non-critical data	Predict location of teams based on mission plan Provide connectivity map to help the user reconnect	Prioritize synchronization of critical messages Eliminate redundant messages

Extensions to the existing DTN standard for priorities, staleness, replacement, and redundancy monitoring to increase bandwidth efficiency in DIL environments

Metadata
· Time and location
· Priority
· Type of payload (image, voice, video, text, …)
· Set of tags describing payload content (building, crowd, fire, injured person, …)

Secure Service VM Migration

Cloudlet Admins exchange temporary keys using their radios

Cloudlet A
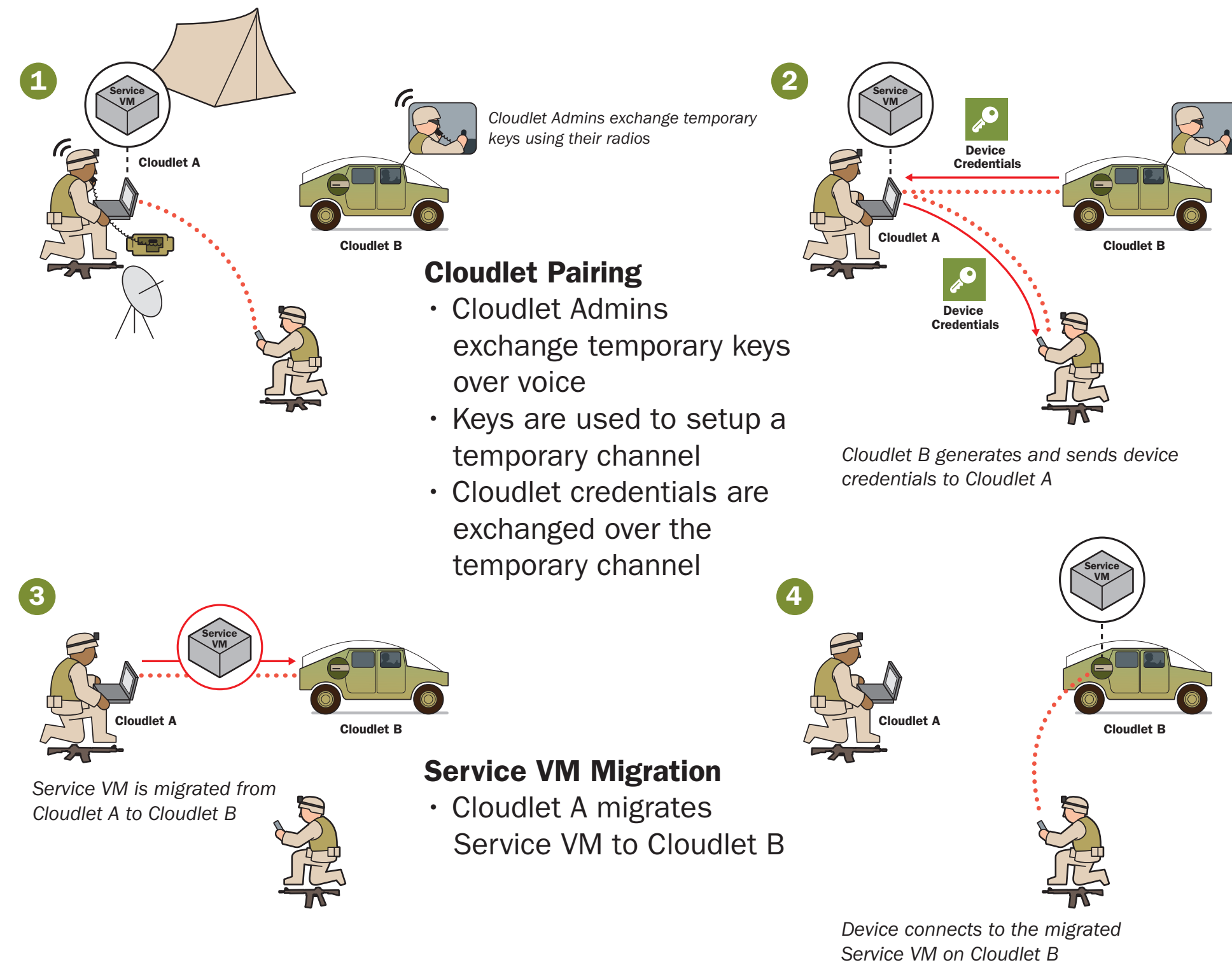
Cloudlet B

Cloudlet Pairing
· Cloudlet Admins exchange temporary keys over voice
· Keys are used to setup a temporary channel
· Cloudlet credentials are exchanged over the temporary channel

Device Credentials

Cloudlet A

Cloudlet B

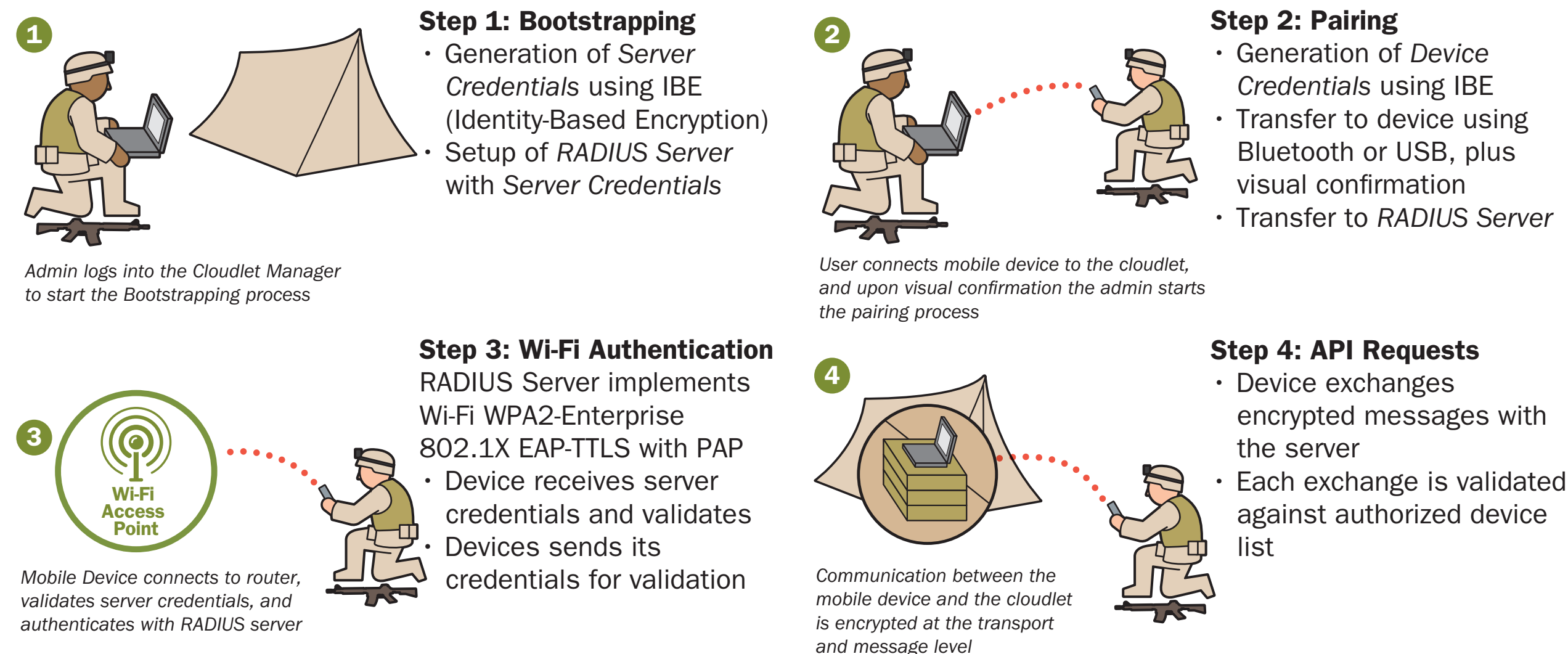Cloudlet B generates and sends device credentials to Cloudlet A

Device Credential Generation
· Cloudlet A discovers and connects to Cloudlet B using exchanged credentials
· Cloudlet B generates new credentials for Device
· Cloudlet B sends credentials to Device via Cloudlet A

Cloudlet A

Cloudlet B

Service VM is migrated from Cloudlet A to Cloudlet B

Service VM Migration
· Cloudlet A migrates Service VM to Cloudlet B

Cloudlet A

Cloudlet B

Device connects to the migrated Service VM on Cloudlet B

Device Connection
· Device connects to Cloudlet B using new credentials
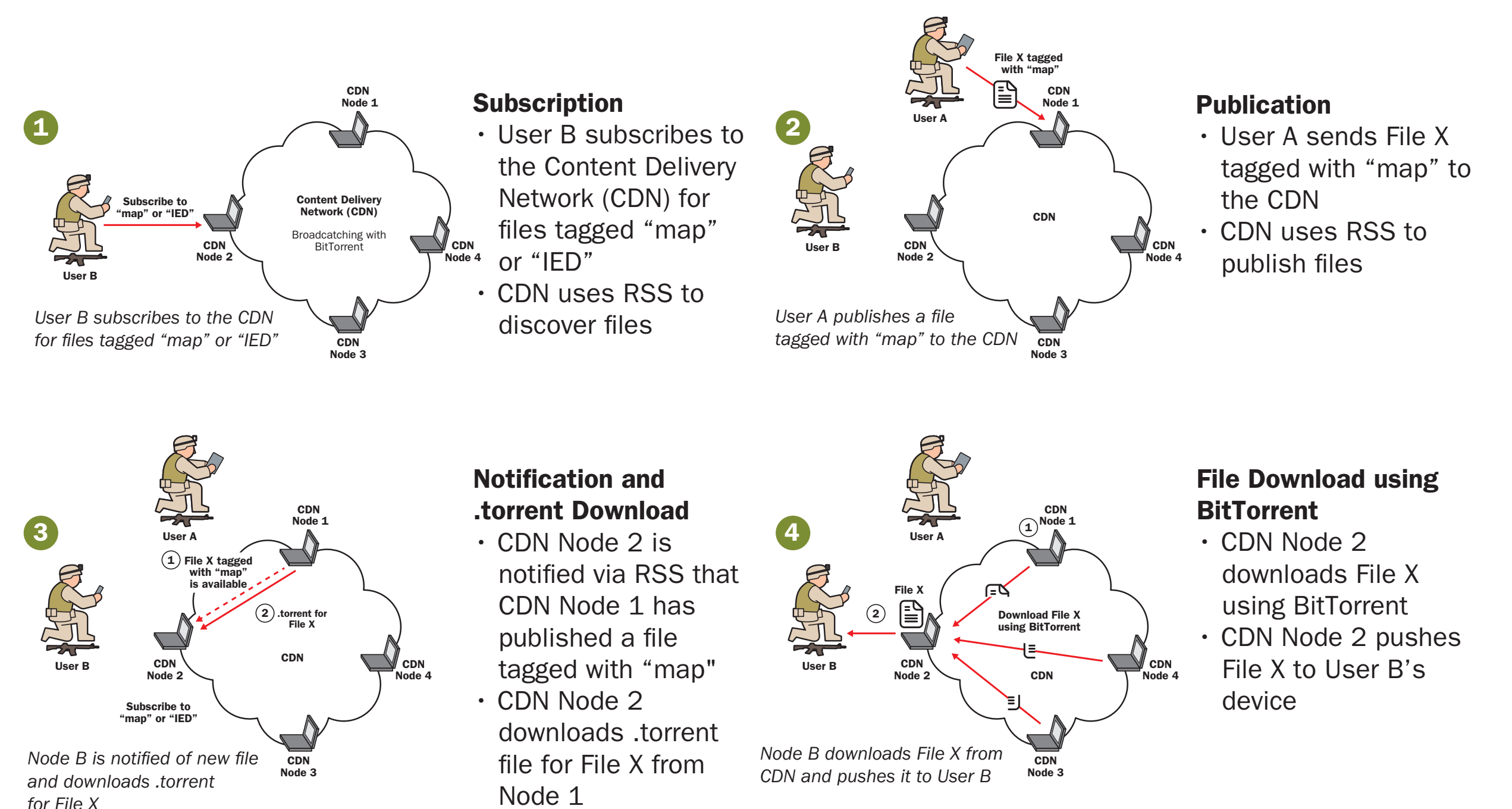· Client App on Device connects to Service VM running on Cloudlet B

Trusted Identities in Disconnected Environments

Step 1: Bootstrapping
· Generation of *Server Credentials* using IBE (Identity-Based Encryption)
· Setup of *RADIUS Server* with *Server Credentials*

Admin logs into the Cloudlet Manager to start the Bootstrapping process

Step 2: Pairing
· Generation of *Device Credentials* using IBE
· Transfer to device using Bluetooth or USB, plus visual confirmation
· Transfer to RADIUS Server

User connects mobile device to the cloudlet, and upon visual confirmation the admin starts the pairing process

Step 3: Wi-Fi Authentication
RADIUS Server implements Wi-Fi WPA2-Enterprise 802.1X EAP-TTLS with PAP
· Device receives server credentials and validates
· Devices sends its credentials for validation

Wi-Fi Access Point

Mobile Device connects to router, validates server credentials, and authenticates with RADIUS server

Step 4: API Requests
· Device exchanges encrypted messages with the server
· Each exchange is validated against authorized device list

Communication between the mobile device and the cloudlet is encrypted at the transport and message level

Device Credential Revocation
· Automatic due to timeout: Bootstrapping requires setting up mission length
· Manual due to known loss or compromise: Cloudlet Manager component has revocation option

Delay-Tolerant Data Sharing

CDN Node 1

CDN Node 2

CDN Node 3

CDN Node 4

User B

Subscribe to "map" or "IED"

Content Delivery Network (CDN) Broadcatching with BitTorrent

User B subscribes to the CDN for files tagged "map" or "IED"

Subscription
· User B subscribes to the Content Delivery Network (CDN) for files tagged "map" or "IED"
· CDN uses RSS to discover files

File X tagged with "map"

User A

CDN Node 1

CDN Node 2

CDN Node 3

CDN Node 4

User A publishes a file tagged with "map" to the CDN

Publication
· User A sends File X tagged with "map" to the CDN
· CDN uses RSS to publish files

User A

CDN Node 1

1. File X tagged with "map" is available

2. .torrent for File X

User B

CDN Node 2

CDN Node 3

CDN Node 4

Subscribe to "map" or "IED"

Node B is notified of new file and downloads .torrent for File X

Notification and .torrent Download
· CDN Node 2 is notified via RSS that CDN Node 1 has published a file tagged with "map"
· CDN Node 2 downloads .torrent file for File X from Node 1

User A

CDN Node 1

File X

Download File X using BitTorrent

User B

CDN Node 2

CDN Node 3

CDN Node 4

Node B downloads File X from CDN and pushes it to User B

File Download using BitTorrent
· CDN Node 2 downloads File X using BitTorrent
· CDN Node 2 pushes File X to User B's device