

# IoT Authentication

CPS and IoT Security

*Alessandro Brighente*

*Master Degree in Cybersecurity*



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA



SPRITZ  
SECURITY & PRIVACY  
RESEARCH GROUP




- Authentication usually assumes that keys are secret and cannot be obtained by adversaries
- Closed network, where only devices that gain a secret through a secure out-of-band channel get their secrets, it might be challenging for attackers
- Among, the others, Zigbee is one of the protocols that has been considered to be secure due its closeness



- There are two factors that makes it challenging to compromise Zigbee networks
- Closed nature: Zigbee devices are equipped with a dedicated commissioning process to add new devices to the network
- Commissioning usually requires users' actions to enable the controller to accept joining requests (e.g., pushing a button on the controller)
- Except commissioning, the Zigbee network is closed and the controller will not process the joining request



- It consists of five layers:
  - Physical
  - Medium Access Control (MAC)
  - Network (NWK)
  - Application Support Sublayer (APS)
  - Zigbee Cluster Library (ZCL)

- It consists of five layers:
    - Physical  Sets one of the 16 channels from 802.15.4
    - Medium Access Control (MAC)
    - Network (NWK)
    - Application Support Sublayer (APS)
    - Zigbee Cluster Library (ZCL)
  - Data transmission and routing
  - Application
- Every coordinator has a 16-bit PAN ID and a 64-bit Extended PAN ID (EPID), which both uniquely defines the network
  - Expressed in clear format in MAC and Network layers

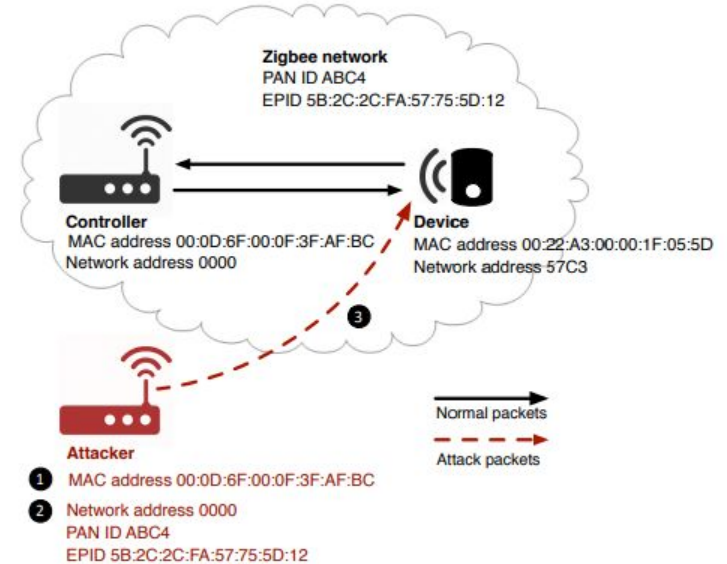


- AES encryption and CCM mode of operation (authenticity and confidentiality)
- 32-bit Message Integrity Code (MIC) is calculated and appended after the encrypted payload for integrity protection
- Brute forcing a 32-bit MIC is infeasible in terms of time consumption
- With a packet transmission rate of 200 packets/s the brute force would take more than 248 days



- Factors making Zigbee networks challenging to compromise:
  - Closed network with a dedicated commissioning process to add new devices in the network (press a button on the controller)
  - Zigbee uses encryption with AES CCM. Without keys you cannot infiltrate the network
- We now want to understand whether it is possible to still infiltrate the network

- Simplified Zigbee network with two nodes: one device, one controller
- Attacker not authorized, not part of the network
- Attacker sniffs publicly available Zigbee network information: MAC, net addresses, PAN ID, EPID (unencrypted in MAC and network layer headers)







- The attacker impersonates a node that is already in the target Zigbee network
- Since the controller has the most capabilities, we focus on impersonation of the controller
- The following attack steps can be launched at arbitrary time during the closed normal operations of Zigbee networks



- Step 1: the attack device needs to overwrite its manufacturer-produced physical address and pretend to be the controller
- This controller's address can be obtained by sniffing Zigbee packets, since the MAC address is contained in plaintext in the header



- Step 2: The attacker further imitates the network identifiers
- Extracts the controller network address and network PAN ID by eavesdropping regular Zigbee packets
- To get EPID, the adversary broadcasts a beacon request
- The controller will send a beacon reply with EPID and state that the network is closed and does not accept join requests
- The adversary selects a target device and obtain its address via packet sniffing



- Step 3: the attack device constructs packets and injects them into the Zigbee network
- The goal is to cause the target device to process forged control packets and end up in dysfunctional statuses
- Though Zigbee uses encryption on the network layer payload, packets crafted with specific control fields and commands can induce vulnerabilities



- We now manipulate MAC packets during the power on phase
- Every time a device boots up, it uses its manufacturer-provided MAC
- We can simply change the code, and replace the one fetched from one of sources including non-volatile memory, flash, or random generation



- More challenging, as the adversary needs to interact with a closed network
- During normal operations the network does not accept association requests, and authorized nodes have their roles in the network
- A new device is not recognized and cannot play any role
- Instead of trying to access the network, we can impersonate the controller and create a new network, as a twin of the original one
- We exploit the Zigbee network formation to enforce network-setup manipulation



- Network formation needs three steps from a controller
  - Enabling the radio antenna
  - Setting PAN ID and EPID
  - Adding routing path of the target device without commissioning
- To avoid the original controller to interfere with the network formation, we assume that the setup is conducted outside of the transmission range of the target Zigbee network (after needed sniffing)



- The 16-bit PAN ID is contained in every Zigbee packet to identify the associated network
- We aim at modifying the PAN ID of the new Zigbee network that the module (attacker) creates and make it the same as the target network
- The PAN ID is stored in a configuration file that can hence be changed upon sniffing the target one





- Zigbee mainly uses PAN ID in regular communications, while EPID is used in commissioning or rejoining process
- An adversary can acquire the EPIDs of the nearby Zigbee networks by simply broadcasting a beacon request
- During network formation, we assign the EPID to the newly created network by setting the specific value



- Although we now have a twin network with the proper identifiers, it does not contain any node yet
- We need to induce the framework module to believe that a target device is associated with this new network and stand-by for communications
- At the end of the network formation, the module will be in open mode for a couple of hundred seconds to accept new nodes to join
- However, a device has no reason to actively sending association requests



- Now that we created a malicious network, we can start exploring whether it is possible to send packets that cause malfunctioning
- We first need to understand what is the structure of packets we can send without legitimate keys
- Then we can start exploring how to randomly generate packets to find vulnerabilities



- A first approach would be to randomly put content into the generated packets and blindly test whether they cause the Zigbee network to malfunction
- However, this is highly inefficient and results in many non valid packets
- Two challenges that we want to address in the fuzzing process:
  - Zigbee uses encryption
  - Packets have varied length and formats according to header values



- With our attack, the malicious coordinator does not know the key
- Therefore, the packets it generates cannot be accepted unless being encrypted with the proper key
- We hence examine unencrypted fields in Zigbee packets like MAC or network headers
- We want to transmit plaintext messages that get processed



- To have the payload of a layer encrypted, there are three security-related fields on that layer: security enabled bit, security AUX header, and message integrity code
- The security bit plays a decisive role: if set to 0, no security mechanism, so no AUX header, nor MIC → forged packets are processed at the receiver!
- Packets with security bit to 1 may have impact if the system has implementation flaws
- We can develop strategies to fuzz this



- If security bit is set to 1, use AES with CCM
- The 128-bit AES-CCM allows generating encryption output with arbitrary length
- AES encrypts an incremental nonce and then XOR with plaintext
- The ciphertext maintains the same length as the plaintext



- In addition to encryption, Zigbee has varied packet formats
- First, the different header values cause different header lengths and consequently change the packet structure
- Second, in the payload, commands and attribute parameters are correlated and require different lengths
- For instance, attribute IDs have different data types to achieve various functionalities
- We hence need to actively enumerate each individual field by sending packets and examining the format changes of captured ones





- Two steps for Zigbee analysis:
  - Decide the header fields to find the fuzzing locations
  - Retrieve the fuzzing ranges of commands and parameters
- First, every layer header has a frame control field which decides the other header fields and whether the packet contains upper layers
- We enumerate the frame control values bit by bit to find the corresponding structure changes and later construct protocol-compliant lower layers when we fuzz the upper layers



- Second, we focus on cluster ID, command ID, and attribute ID in different upper layers for our fuzzing
- These fields are correlated across different layers
- For example, different command IDs require different lengths of attribute IDs, and the cluster ID in the APS header determines the ZCL layer command ID
- We test their minimum and maximum values to get ranges



- We first configure the network layer header and payload
- The header can use three types of address settings (mix of network and MAC addresses) and two security bit settings (0, 1)
- Network layer packets include command ID, and each command has a one byte attribute that can be fuzzed
- There are 13 valid network commands
- If security bit = 0,  $13 \times 2^8 = 3328$  combinations to try



- If security bit =1, network payload is encrypted and MIC added for integrity check
- Since encryption preserves the content length, we prioritize cases that could bring to meaningful results
- All network commands are 1 byte, and add another byte as attributes
- We set random MIC values and fuzz the payload only with the length of possible commands



- We fuzz encryption payload lengths of 8 and 16 bits  $\rightarrow 2^8 + 2^{16} = 65792$  combinations
- NWK header has 3 unencrypted bits that can be fuzzed for different packet settings
- Total fuzzing number is  $65792 \times 2^3 = 526336$



- This attack cause vulnerable systems to leak security information
- Some of the ZCL cluster ID fuzzed can cause key leakage in Zigbee networks
- The attack packet will cause the device to send rejoin requests
- During the rejoin process, the controller will resend the network key which is only encrypted by the default public link key
- With the publicly known link key and the frame counter in the packet (plaintext), we are able to decrypt and retrieve the network key of the network



# Group Pairing and Key Exchange



- Traditional approach for key management envision a central device with authoritative capabilities handling them all
- However, this represents a single point of failure that may compromise the overall network security
- IoT platforms have recently been pushing towards decentralized IoT networking protocols (e.g., OpenThread)
- Past efforts at decentralized IoT device pairing include two approaches: human-in-the-loop, and context-based pairing



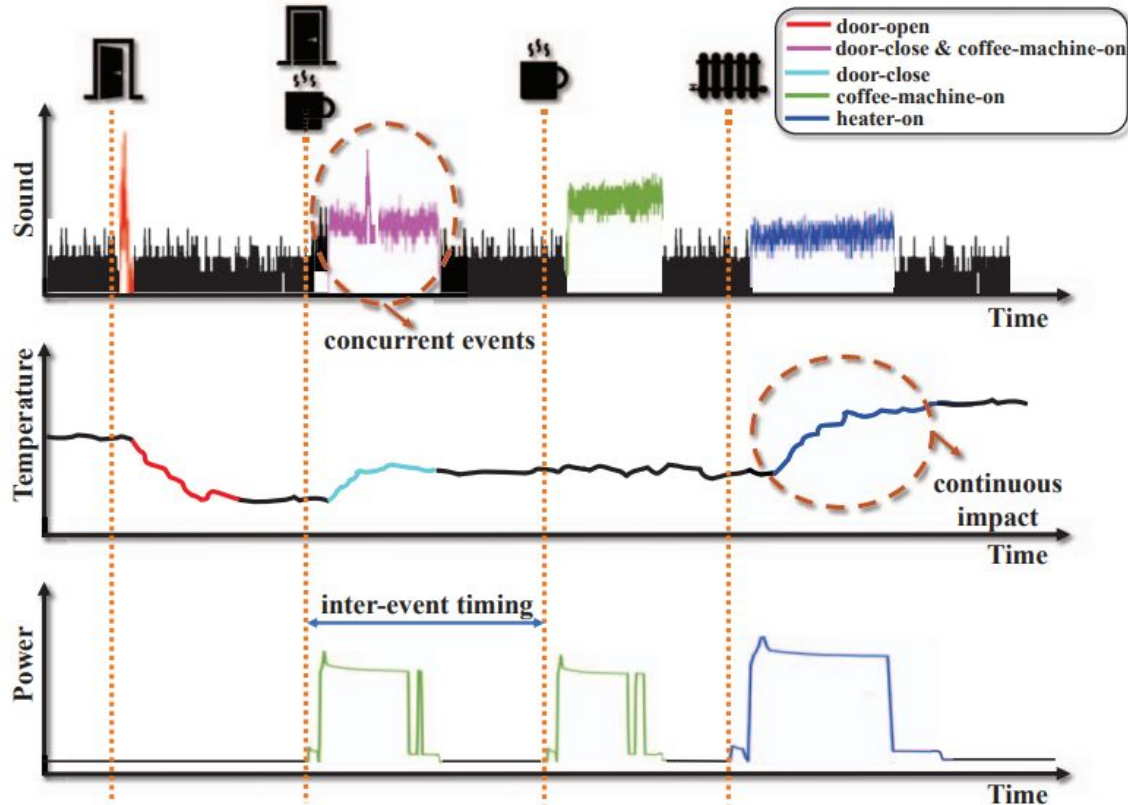


- In human-in-the-loop approaches a human needs to be physically involved in the pairing process
- For instance, the user should touch or press a button, shake two devices at the same time, enter password, or read QR codes
- With context based pairing we can increase scalability, as the human is not needed
- In this, co-located sensors establish shared keys based on the entropy extracted when they observe common events



- We consider an IoT deployment with three devices
- Each device is equipped with either a microphone, a power meter, a temperature sensor
- User A opens the door to go aìout, in the meanwhile user B turns on the coffee machine
- While the coffee machine is on, user A returns and closes the door
- User A prepares a cup of coffee for herself and turns on the heater

# An Example of Events



# An Example of Events



<b>Event</b>	<b>Sensors Impacted</b>
door-open/close	air pressure, humidity, illuminance, microphone, motion, temperature
coffee-machine-on/off	microphone, power
window-open/close	air pressure, humidity, illuminance, motion, temperature
oven-on/off	humidity, power, temperature
light-on/off	illuminance, power
AC-on/off	air pressure, humidity, microphone, power, temperature
heater-on/off	humidity, microphone, power, temperature
TV-on/off	illuminance, microphone, power
dryer-on/off	humidity, microphone, power, temperature



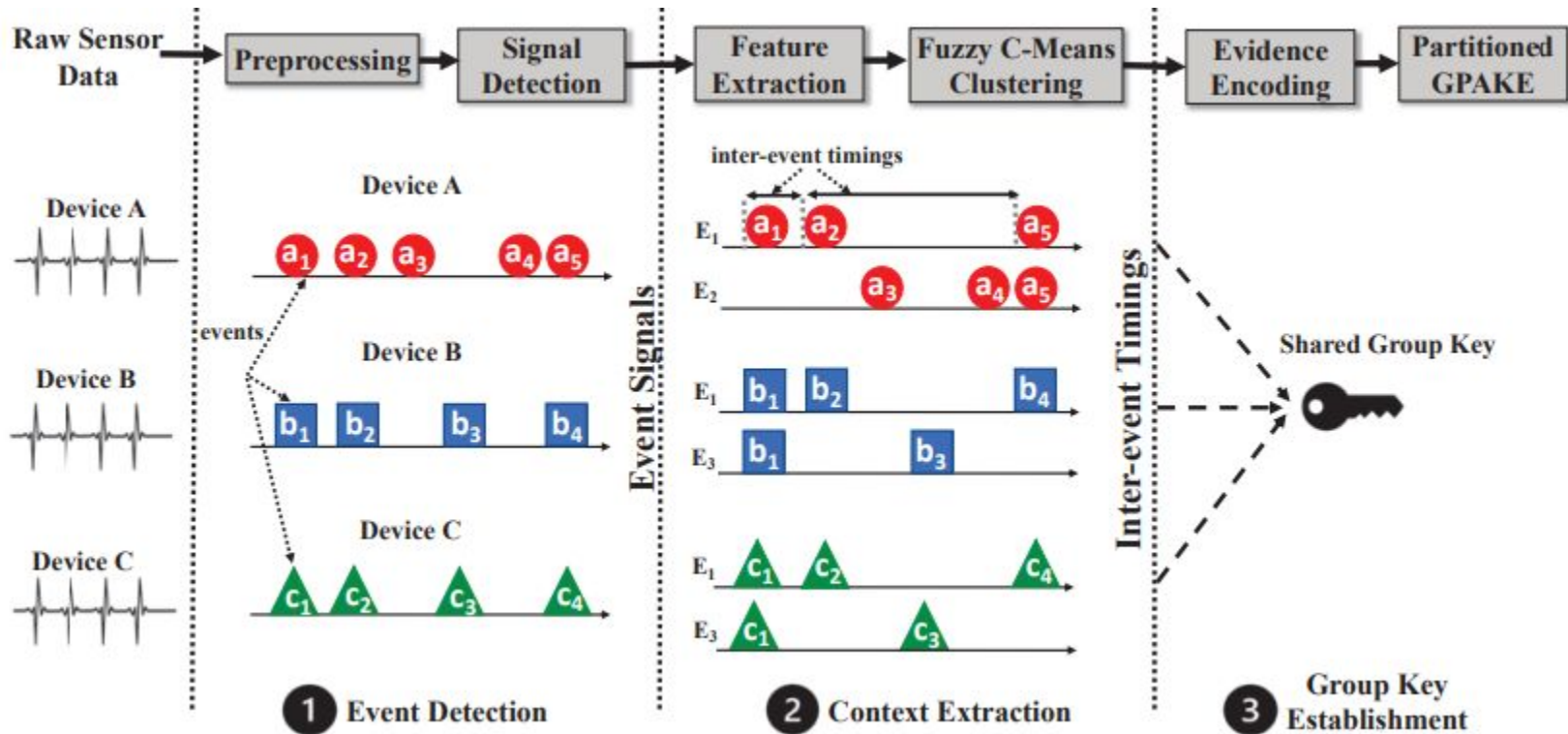
- We consider an attacker aiming at eavesdropping the communication between IoT devices and learn private information about users
- Devices are deployed within an indoor closed physical space and controlled by a common trusted entity
- The attacker is not present within the boundaries of the indoor IoT environment and cannot access, add devices or control devices inside the network
- The attacker has complete knowledge of the pairing protocol and has access to the communication channels



- We now present IoT Cupid, a solution for context-based IoT device pairing
- The first step is to process the raw time-series data collected in real-time and perform a threshold-based signal detection to separate the sensor data corresponding to events from background noise
- The second step is to extract distinctive time-series features from the signals each sensor has detected
- It then extends a fuzzy clustering algorithm to group independent and concurrent signals into different events



- The clustered events are used to obtain the sequence of time intervals between consecutive events of a given type, serving as evidence of the device's context
- Lastly, IoT devices use their inter-event timings to authenticate each other and establish a shared group key
- IoT Cupid encodes the inter-event timings into passwords and extends a partitioned group password-based authenticated key exchange scheme for a group key establishment protocol







- We consider devices that sense the same event as a *group*
- Each subset of devices that have the same inter-event timings establishes a group key
- IoT Cupid does not require a central gateway or IoT hub, but guarantees secure ad-hoc connectivity among heterogeneous IoT devices
- To initiate association, devices broadcast their public keys encrypted with the extracted inter-event timing for establishing group keys



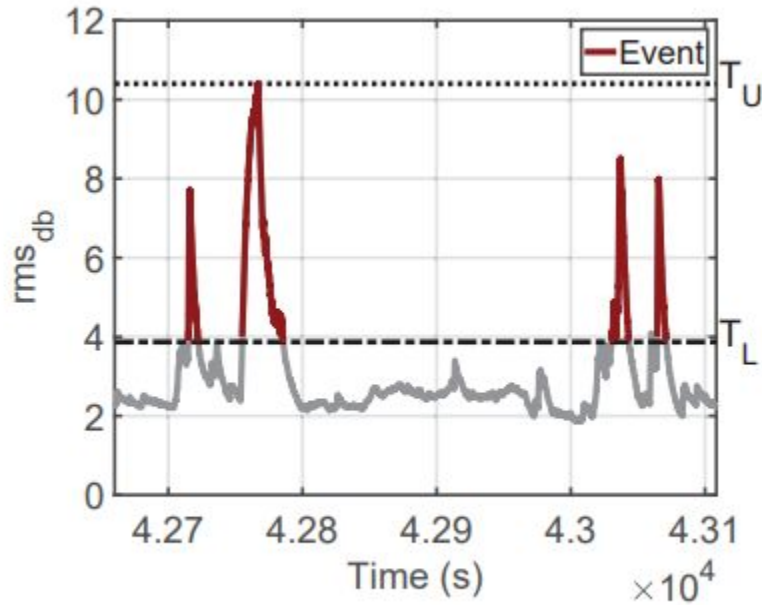
- To extract signals corresponding to events, we first segment sensor data into multiple samples with window size  $w_s$
- To address fluctuations during the day, we normalize the sensor readings to eliminate these fluctuations impact and capture transient changes caused by events
- We then apply a smoothing filter by computing the exponentially weighted moving average

$S_w = a * Y_w + (1-a) * S_{w-1}$ , where  $a$  is the weight,  $Y_w$  is the sensor data and  $S_{w-1}$  is the EWMA of the preceding window

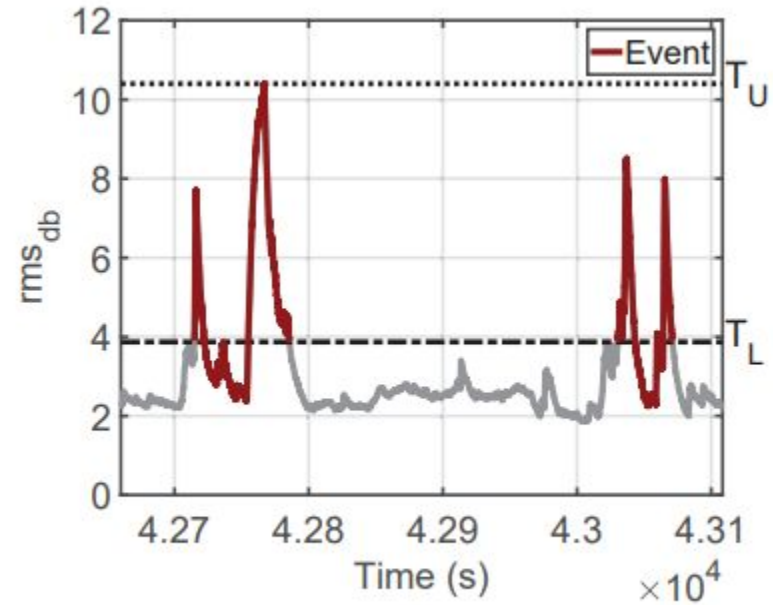


- We use a threshold-based approach to distinguish event's influence on sensor readings from background noise
- We use a lower threshold  $T_L$  to identify peaks in sensor readings that distinguish events' impact from background noise
- We use an upper threshold  $T_U$  to remove high amplitude noise signals
- We consider the consecutive timestamps at which sensor values exceed  $T_L$  but are below  $T_U$  as a single event

# Event Detection



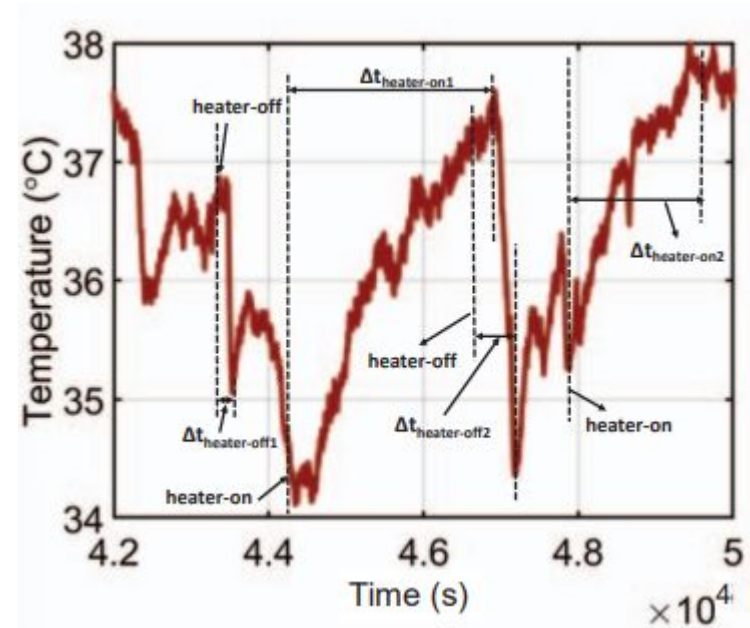
(a)



(b)

Discontinuities aggregated to avoid a single event being classified as multiple

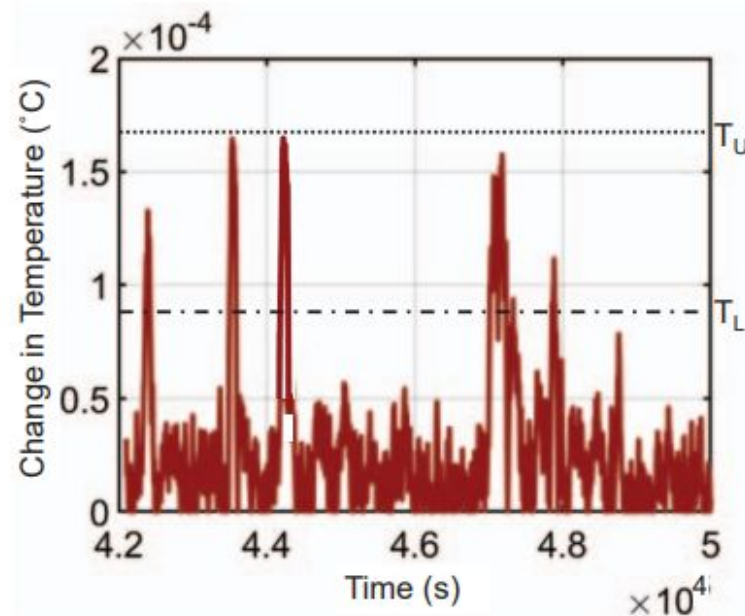
- If sensors measure continuous quantities (e.g., temperature), the previous approach is not good
- For instance, a heater-on event occurring at time  $t$  causes a gradual increase in temperature sensor values after a delay  $\Delta t_t$
- Same event at different timing may have different delays (figure)





- To account for gradual changes and the varying delay, we leverage the rate of change in the sensor readings to detect signals corresponding to events for continuously influenced sensors
- We first compute the derivative of the pre-processed sensor values in each window  $w$  as  $S'_w = (S_{w_{ws}} - S_{w_0})/w_s$ , where  $w_s$  is the window size and the terms in parentheses are recorded as the first and last sensor values in the window
- We then apply lower and upper thresholds based on the average derivative of each sensor

We extract the timestamps where the absolute value of the sensor readings' derivatives lie within the predetermined  $T_U$  and  $T_L$  for the sensor

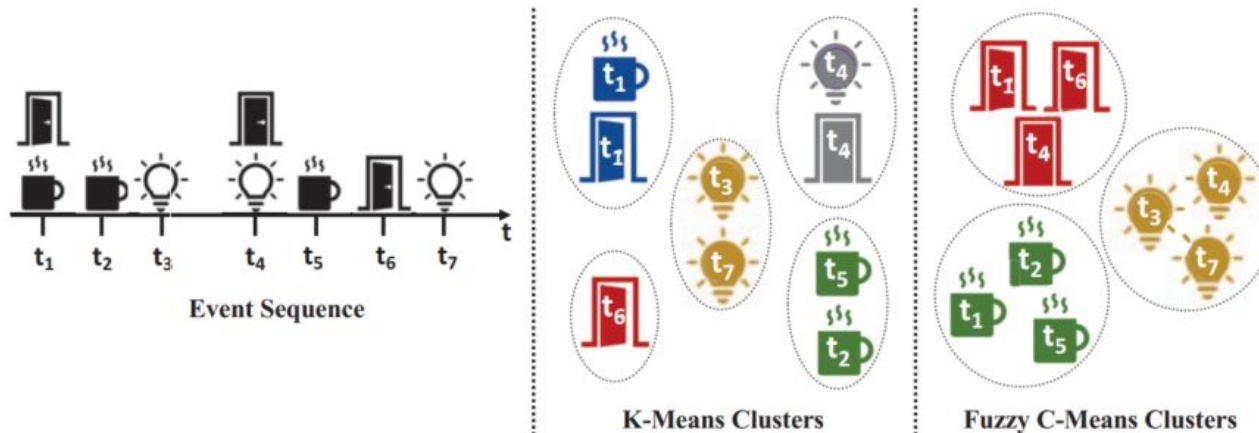




- Context is about event clustering
- The idea is to cluster the detected signals into different events to extract their inter-event timings
- We do not leverage any prior information about the event types for clustering, as a single device might detect multiple events
- As a first step, we extract **time domain features** (min, max,..) from the signals corresponding to events
- To select a set of features  $F_{\min}$  we perform dimensionality reduction via Principal Component Analysis (PCA)



- In real deployments, multiple events may occur simultaneously and produce overlapping signals
- This implies that the signal features of simultaneous events might significantly differ from those in single events



Hard clustering is not a good idea in case of simultaneous events



- We extend fuzzy C-Means clustering to assign the detected signals into one or more appropriate event clusters based on the extracted features
- We partition signals into  $c$  (input parameter) clusters
- This method allow events occurring simultaneously to belong to their appropriate cluster



- IoT Cupid generates inter-event timings for each device after clustering the detected signals into different event types
- Each device extracts inter-event times of event occurrences in each cluster, and uses it as evidence in the group key establishment
- With this, although devices may be heterogeneous, two devices detecting the same event will measure the same inter-event timings
- Although devices might be non-synchronized, inter-event timings are



- Recalling that the IoT deployment might be dynamic, deriving group keys might be challenging
- First, groups must be generated dynamically based on the devices that sense the same event
- Second, the protocol should support device addition and removal
- When a device is added, it should pair with other nodes
- When a device is removed its keys must be revoked to prevent an adversary from capturing them



- Group keys can be generated using the secure communication channels from individual keys derived through a standard pairing protocol
- Group Diffie-Hellman is one of them, but it requires multiple rounds
- Furthermore, when a device is added to the IoT network, it must first individually pair with other devices to be authenticated and then participate in the group key establishment
- NOT A GOOD OPTION



- We could use fuzzy commitment schemes to generate individual keys
- The idea is to use error-correcting codes and enable verifying two evidences when they have small differences (Hamming distance  $< th.$ )
- However, these schemes are vulnerable to *offline brute-force key guessing attacks*
- The adversary collects the network traffic and tries all evidences until they find the one that can decrypt the network traffic



- To prevent offline brute-force key guessing attacks, we could use a large number of evidences (i.e., a large number of inter-arrival times)
- This is however highly inefficient, as it may take a long time to derive keys
- Password-Authenticated Key Exchange (PAKE) protocols have been proposed to prevent offline brute-force attacks
- We should however extend them into the group setting



- GPAKE enables multiple devices sharing the same evidence to derive group keys
- The passwords of all devices that participate in the key agreement must be the same, as the scheme abort without establishing a shared key even if a single password is different
- An adversary could leverage this limitation by joining the key agreement protocol with arbitrary evidences to deny legitimate key derivation → *denial of key exchange*





- Objectives: dynamic group generation with computational efficiency, device addition/removal, resilience to offline brute-force and denial of key exchange attacks
- We encode inter-event timings to be used as passwords
- Devices then use the passwords to run a partitioned GPAKE scheme such that each subset of devices sensing the same events derives a group key

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
	<b>Step 1: Evidence Extraction</b>			
①	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$	$\{i_{1,d_2} \dots i_{c_2,d_2}\} = \text{CONTEXT\_EXTRACTION}(d_2)$		$\{i_{1,d_N} \dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
	<b>Step 2: Encoding</b>			
②	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \lfloor \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \rfloor$	$\{pw_{1,d_2} \dots pw_{c_2,d_2}\} = \lfloor \{i_{1,d_2} \dots i_{c_2,d_2}\} / W \rfloor$		$\{pw_{1,d_N} \dots pw_{c_N,d_N}\} = \lfloor \{i_{1,d_N} \dots i_{c_N,d_N}\} / W \rfloor$
	<b>Step 3: Partitioned GPAKE</b>			
③	Determine the public parameters, two primes $p$ and $q$ , a finite field $\mathbb{F}_q$ and a group $\mathbb{Z}_p$ . $E(\mathbb{F}_q)$ is an elliptic curve, and $P \in E(\mathbb{F}_q)$ is its generator.			
④	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$	Choose random $\{x_{1,d_2} \dots x_{c_2,d_2}\} \xleftarrow{\$} \mathbb{Z}_p$		Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
⑤	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \text{ mod } q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$	$X_{i,d_2} \leftarrow x_{i,d_2} \cdot P \text{ mod } q, Y_{i,d_2} \leftarrow \text{Enc}_{pw_{i,d_2}}(X_{i,d_2})$		$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \text{ mod } q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
⑥	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$	Broadcast $(d_2, Y_{i,d_2})$ , where $i \in \{1, \dots, c_2\}$		Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
⑦		For every received message $(d_j, Y_{j,d_j})$ , where $j \in \{1, \dots, N\}$ :		
⑧	$X_{j,d_j} \leftarrow \text{Dec}_{pw_{j,d_1}}(Y_{j,d_j})$ , if $(X_{j,d_j} \in E(\mathbb{F}_q))$ :	$X_{j,d_j} \leftarrow \text{Dec}_{pw_{j,d_2}}(Y_{j,d_j})$ , if $(X_{j,d_j} \in E(\mathbb{F}_q))$ :		$X_{j,d_j} \leftarrow \text{Dec}_{pw_{j,d_N}}(Y_{j,d_j})$ , if $(X_{j,d_j} \in E(\mathbb{F}_q))$ :
⑨	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_j}\}$	$\text{sid}_{i,j,d_2} = \{d_2, Y_{i,d_2}, d_j, Y_{j,d_j}\}$		$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_j}\}$
⑩	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_j}, x_{i,d_1} \cdot X_{j,d_j} \text{ mod } q)$	$\text{sk}_{i,j,d_2} \leftarrow H(d_2, d_j, X_{i,d_2}, X_{j,d_j}, x_{i,d_2} \cdot X_{j,d_j} \text{ mod } q)$		$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_j}, x_{i,d_N} \cdot X_{j,d_j} \text{ mod } q)$
⑪	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
⑫	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
⑬	For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :			
⑭	if $(Y_{i,d_1} \in \text{sid}_{i,j,d_j})$ : $r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_j})$	if $(Y_{i,d_2} \in \text{sid}_{i,j,d_j})$ : $r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_j})$		if $(Y_{i,d_N} \in \text{sid}_{i,j,d_j})$ : $r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_j})$
⑮	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
	<b>Step 1: Evidence Extraction</b>			
①	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$	$\{i_{1,d_2} \dots i_{c_2,d_2}\} = \text{CONTEXT\_EXTRACTION}(d_2)$		$\{i_{1,d_N} \dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
	<b>Step 2: Encoding</b>			
②	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \lfloor \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \rfloor$	<b>Inter-event timing extraction</b>		$\{pw_{1,d_N} \dots pw_{c_N,d_N}\} = \lfloor \{i_{1,d_N} \dots i_{c_N,d_N}\} / W \rfloor$
③	Determine the public parameters, $(q, P)$ is an elliptic curve, and $P \in E(\mathbb{F}_q)$ is its generator.			
④	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$			Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
⑤	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \text{ mod } q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$			$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \text{ mod } q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
⑥	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$	Broadcast $(d_2, Y_{i,d_2})$ , where $i \in \{1, \dots, c_2\}$		Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
⑦		For every received message $(d_j, Y_{j,d_j})$ , where $j \in \{1, \dots, N\}$ :		
⑧	$X_{j,d_1} \leftarrow \text{Dec}_{pw_{j,d_1}}(Y_{j,d_j})$ , if $(X_{j,d_1} \in E(\mathbb{F}_q))$ :	$X_{j,d_2} \leftarrow \text{Dec}_{pw_{j,d_2}}(Y_{j,d_j})$ , if $(X_{j,d_2} \in E(\mathbb{F}_q))$ :		$X_{j,d_N} \leftarrow \text{Dec}_{pw_{j,d_N}}(Y_{j,d_j})$ , if $(X_{j,d_N} \in E(\mathbb{F}_q))$ :
⑨	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_j}\}$	$\text{sid}_{i,j,d_2} = \{d_2, Y_{i,d_2}, d_j, Y_{j,d_j}\}$		$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_j}\}$
⑩	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_j}, x_{i,d_1} \cdot X_{j,d_j} \text{ mod } q)$	$\text{sk}_{i,j,d_2} \leftarrow H(d_2, d_j, X_{i,d_2}, X_{j,d_j}, x_{i,d_2} \cdot X_{j,d_j} \text{ mod } q)$		$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_j}, x_{i,d_N} \cdot X_{j,d_j} \text{ mod } q)$
⑪	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
⑫	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
⑬		For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :		
⑭	if $(Y_{i,d_1} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_1} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_j})$	if $(Y_{i,d_2} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_2} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_j})$		if $(Y_{i,d_N} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_N} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_j})$
⑮	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
	<b>Step 1: Evidence Extraction</b>			
①	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$	$\{i_{1,d_2} \dots i_{c_2,d_2}\} = \text{CONTEXT\_EXTRACTION}(d_2)$		$\{i_{1,d_N} \dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
	<b>Step 2: Encoding</b>			
②	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \lfloor \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \rfloor$	$\{pw_{1,d_2} \dots pw_{c_2,d_2}\} = \lfloor \{i_{1,d_2} \dots i_{c_2,d_2}\} / W \rfloor$		$\{pw_{1,d_N} \dots pw_{c_N,d_N}\} = \lfloor \{i_{1,d_N} \dots i_{c_N,d_N}\} / W \rfloor$
	<b>Step 3: Partitioned GPAKE</b>			
③	Determine the public parameters, two primes $p$ and $q$ , a finite field $\mathbb{F}_q$ and a group $\mathbb{Z}_p$ . $E(\mathbb{F}_q)$ is an elliptic curve, and $P \in E(\mathbb{F}_q)$ is its generator.			
④	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$			Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
⑤	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \text{ mod } q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$			$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \text{ mod } q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
⑥	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$			Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
⑦				
⑧	$X_{j,d_j} \leftarrow \text{Dec}_{pw_{i,d_1}}(Y_{j,d_j}), \text{ if } (X_{j,d_j} \in E(\mathbb{F}_q)) :$			$X_{j,d_j} \leftarrow \text{Dec}_{pw_{i,d_N}}(Y_{j,d_j}), \text{ if } (X_{j,d_j} \in E(\mathbb{F}_q)) :$
⑨	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_j}\}$			$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_j}\}$
⑩	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_j}, x_{i,d_1} \cdot X_{j,d_j} \text{ mod } q)$			$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_j}, x_{i,d_N} \cdot X_{j,d_j} \text{ mod } q)$
⑪	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$			$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
⑫	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
⑬	For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :			
⑭	$\text{if } (Y_{i,d_1} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_j})$	$\text{if } (Y_{i,d_2} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_j})$		$\text{if } (Y_{i,d_N} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_j})$
⑮	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$

Compensate different operating frequencies using a quantization window  $W \rightarrow$  trade-off efficiency and pwd entropy

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
<b>Step 1: Evidence Extraction</b>				
①	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$	<b>Public EC parameters</b>		$\dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
②	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \lfloor \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \rfloor$	$\{pw_{1,d_2} \dots pw_{c_2,d_2}\} = \lfloor \{i_{1,d_2} \dots i_{c_2,d_2}\} / W \rfloor$		$\{pw_{1,d_N} \dots pw_{c_N,d_N}\} = \lfloor \{i_{1,d_N} \dots i_{c_N,d_N}\} / W \rfloor$
<b>Step 3: Partitioned GPAKE</b>				
③	Determine the public parameters, two primes $p$ and $q$ , a finite field $\mathbb{F}_q$ and a group $\mathbb{Z}_p$ . $E(\mathbb{F}_q)$ is an elliptic curve, and $P \in E(\mathbb{F}_q)$ is its generator.			
④	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$	Choose random $\{x_{1,d_2} \dots x_{c_2,d_2}\} \xleftarrow{\$} \mathbb{Z}_p$		Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
⑤	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \text{ mod } q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$	$X_{i,d_2} \leftarrow x_{i,d_2} \cdot P \text{ mod } q, Y_{i,d_2} \leftarrow \text{Enc}_{pw_{i,d_2}}(X_{i,d_2})$		$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \text{ mod } q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
⑥	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$	Broadcast $(d_2, Y_{i,d_2})$ , where $i \in \{1, \dots, c_2\}$		Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
⑦		For every received message $(d_j, Y_{j,d_j})$ , where $j \in \{1, \dots, N\}$ :		
⑧	$X_{j,d_j} \leftarrow \text{Dec}_{pw_{j,d_1}}(Y_{j,d_j})$ , if $(X_{j,d_j} \in E(\mathbb{F}_q))$ :	$X_{j,d_j} \leftarrow \text{Dec}_{pw_{j,d_2}}(Y_{j,d_j})$ , if $(X_{j,d_j} \in E(\mathbb{F}_q))$ :		$X_{j,d_j} \leftarrow \text{Dec}_{pw_{j,d_N}}(Y_{j,d_j})$ , if $(X_{j,d_j} \in E(\mathbb{F}_q))$ :
⑨	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_j}\}$	$\text{sid}_{i,j,d_2} = \{d_2, Y_{i,d_2}, d_j, Y_{j,d_j}\}$		$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_j}\}$
⑩	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_j}, x_{i,d_1} \cdot X_{j,d_j} \text{ mod } q)$	$\text{sk}_{i,j,d_2} \leftarrow H(d_2, d_j, X_{i,d_2}, X_{j,d_j}, x_{i,d_2} \cdot X_{j,d_j} \text{ mod } q)$		$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_j}, x_{i,d_N} \cdot X_{j,d_j} \text{ mod } q)$
⑪	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
⑫	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
⑬	For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :			
⑭	if $(Y_{i,d_1} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_j})$	if $(Y_{i,d_2} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_j})$		if $(Y_{i,d_N} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_j})$
⑮	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
	<b>Step 1: Evidence Extraction</b>			
①	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$	$\{i_{1,d_2} \dots i_{c_2,d_2}\} = \text{CONTEXT\_EXTRACTION}(d_2)$		$\{i_{1,d_N} \dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
	<b>Step 2: Encoding</b>			
②	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \lfloor \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \rfloor$	$\{pw_{1,d_2} \dots pw_{c_2,d_2}\} = \lfloor \{i_{1,d_2} \dots i_{c_2,d_2}\} / W \rfloor$		$\{pw_{1,d_N} \dots pw_{c_N,d_N}\} = \lfloor \{i_{1,d_N} \dots i_{c_N,d_N}\} / W \rfloor$
	<b>Step 3: Partitioned GPAKE</b>			
③	Determine the public parameters, two primes $p$ and $q$ , a finite field $\mathbb{F}_q$ and a group $\mathbb{Z}_p$ . $E(\mathbb{F}_q)$ is an elliptic curve, and $P \in E(\mathbb{F}_q)$ is its generator.			
④	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$	Choose random $\{x_{1,d_2} \dots x_{c_2,d_2}\} \xleftarrow{\$} \mathbb{Z}_p$		Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
⑤	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \text{ mod } q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$	$X_{i,d_2} \leftarrow x_{i,d_2} \cdot P \text{ mod } q, Y_{i,d_2} \leftarrow \text{Enc}_{pw_{i,d_2}}(X_{i,d_2})$		$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \text{ mod } q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
⑥	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$			Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
⑦	Generate key pair per event type, encrypt pk with pwd			
⑧				
⑨	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_j}\}$			$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_j}\}$
⑩	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_j}, x_{i,d_1} \cdot X_{j,d_j} \text{ mod } q)$			$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_j}, x_{i,d_N} \cdot X_{j,d_j} \text{ mod } q)$
⑪	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
⑫	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
⑬	For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :			
⑭	$\text{if } (Y_{i,d_1} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_j})$	$\text{if } (Y_{i,d_2} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_j})$		$\text{if } (Y_{i,d_N} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_j} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_j})$
⑮	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
<b>Step 1: Evidence Extraction</b>				
①	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$	<b>Broadcast encrypted pk with id</b>		$\dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
②	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \lfloor \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \rfloor$	$\{pw_{1,d_2} \dots pw_{c_2,d_2}\} = \lfloor \{i_{1,d_2} \dots i_{c_2,d_2}\} / W \rfloor$		$\{pw_{1,d_N} \dots pw_{c_N,d_N}\} = \lfloor \{i_{1,d_N} \dots i_{c_N,d_N}\} / W \rfloor$
<b>Step 3: Partitioned GPAKE</b>				
③	Determine the public parameters, two primes $p$ and $q$ , a finite field $\mathbb{F}_q$ and a group $\mathbb{Z}_p$ . $E(\mathbb{F}_q)$ is an elliptic curve, and $P \in E(\mathbb{F}_q)$ is its generator.			
④	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$	Choose random $\{x_{1,d_2} \dots x_{c_2,d_2}\} \xleftarrow{\$} \mathbb{Z}_p$		Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
⑤	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \text{ mod } q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$	$X_{i,d_2} \leftarrow x_{i,d_2} \cdot P \text{ mod } q, Y_{i,d_2} \leftarrow \text{Enc}_{pw_{i,d_2}}(X_{i,d_2})$		$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \text{ mod } q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
⑥	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$	Broadcast $(d_2, Y_{i,d_2})$ , where $i \in \{1, \dots, c_2\}$		Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
⑦		For every received message $(d_j, Y_{j,d_j})$ , where $j \in \{1, \dots, N\}$ :		
⑧	$X_{j,d_1} \leftarrow \text{Dec}_{pw_{j,d_1}}(Y_{j,d_j})$ , if $(X_{j,d_1} \in E(\mathbb{F}_q))$ :	$X_{j,d_2} \leftarrow \text{Dec}_{pw_{j,d_2}}(Y_{j,d_j})$ , if $(X_{j,d_2} \in E(\mathbb{F}_q))$ :		$X_{j,d_N} \leftarrow \text{Dec}_{pw_{j,d_N}}(Y_{j,d_j})$ , if $(X_{j,d_N} \in E(\mathbb{F}_q))$ :
⑨	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_j}\}$	$\text{sid}_{i,j,d_2} = \{d_2, Y_{i,d_2}, d_j, Y_{j,d_j}\}$		$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_j}\}$
⑩	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_j}, x_{i,d_1} \cdot X_{j,d_j} \text{ mod } q)$	$\text{sk}_{i,j,d_2} \leftarrow H(d_2, d_j, X_{i,d_2}, X_{j,d_j}, x_{i,d_2} \cdot X_{j,d_j} \text{ mod } q)$		$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_j}, x_{i,d_N} \cdot X_{j,d_j} \text{ mod } q)$
⑪	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
⑫	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
⑬	For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :			
⑭	if $(Y_{i,d_1} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_1} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_j})$	if $(Y_{i,d_2} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_2} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_j})$		if $(Y_{i,d_N} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_N} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_j})$
⑮	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
<b>Step 1: Evidence Extraction</b>				
①	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$			$\dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
②	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \lfloor \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \rfloor$			$\dots pw_{c_N,d_N}\} = \lfloor \{i_{1,d_1} \dots i_{c_N,d_N}\} / W \rfloor$
③	Determine the public parameters, two p			urve, and $P \in E(\mathbb{F}_q)$ is its generator.
④	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$	Choose random $\{x_{1,d_2} \dots x_{c_2,d_2}\} \xleftarrow{\$} \mathbb{Z}_p$		Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
⑤	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \text{ mod } q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$	$X_{i,d_2} \leftarrow x_{i,d_2} \cdot P \text{ mod } q, Y_{i,d_2} \leftarrow \text{Enc}_{pw_{i,d_2}}(X_{i,d_2})$		$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \text{ mod } q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
⑥	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$	Broadcast $(d_2, Y_{i,d_2})$ , where $i \in \{1, \dots, c_2\}$		Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
⑦		For every received message $(d_j, Y_{j,d_j})$ , where $j \in \{1, \dots, N\}$ :		
⑧	$X_{j,d_1} \leftarrow \text{Dec}_{pw_{i,d_1}}(Y_{j,d_1})$ , if $(X_{j,d_1} \in E(\mathbb{F}_q))$ :	$X_{j,d_2} \leftarrow \text{Dec}_{pw_{i,d_2}}(Y_{j,d_2})$ , if $(X_{j,d_2} \in E(\mathbb{F}_q))$ :		$X_{j,d_N} \leftarrow \text{Dec}_{pw_{i,d_N}}(Y_{j,d_N})$ , if $(X_{j,d_N} \in E(\mathbb{F}_q))$ :
⑨	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_j}\}$	$\text{sid}_{i,j,d_2} = \{d_2, Y_{i,d_2}, d_j, Y_{j,d_j}\}$		$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_j}\}$
⑩	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_j}, x_{i,d_1} \cdot X_{j,d_j} \text{ mod } q)$	$\text{sk}_{i,j,d_2} \leftarrow H(d_2, d_j, X_{i,d_2}, X_{j,d_j}, x_{i,d_2} \cdot X_{j,d_j} \text{ mod } q)$		$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_j}, x_{i,d_N} \cdot X_{j,d_j} \text{ mod } q)$
⑪	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
⑫	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
⑬		For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :		
⑭	if $(Y_{i,d_1} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_1} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_j})$	if $(Y_{i,d_2} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_2} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_j})$		if $(Y_{i,d_N} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_N} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_j})$
⑮	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$

Try decryption with all pwds; if one matches get public key



# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
	<b>Step 1: Evidence Extraction</b>			
①	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$			$\dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
②	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \lfloor \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \rfloor$			$\dots pw_{c_N,d_N}\} = \lfloor \{i_{1,d_1} \dots i_{c_N,d_N}\} / W \rfloor$
③	Determine the public parameters, two p			curve, and $P \in E(\mathbb{F}_q)$ is its generator.
④	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$	Choose random $\{x_{1,d_2} \dots x_{c_2,d_2}\} \xleftarrow{\$} \mathbb{Z}_p$		Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
⑤	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \bmod q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$	$X_{i,d_2} \leftarrow x_{i,d_2} \cdot P \bmod q, Y_{i,d_2} \leftarrow \text{Enc}_{pw_{i,d_2}}(X_{i,d_2})$		$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \bmod q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
⑥	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$	Broadcast $(d_2, Y_{i,d_2})$ , where $i \in \{1, \dots, c_2\}$		Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
⑦		For every received message $(d_j, Y_{j,d_j})$ , where $j \in \{1, \dots, N\}$ :		
⑧	$X_{j,d_1} \leftarrow \text{Dec}_{pw_{i,d_1}}(Y_{j,d_j})$ , if $(X_{j,d_1} \in E(\mathbb{F}_q))$ :	$X_{j,d_2} \leftarrow \text{Dec}_{pw_{i,d_2}}(Y_{j,d_j})$ , if $(X_{j,d_2} \in E(\mathbb{F}_q))$ :		$X_{j,d_N} \leftarrow \text{Dec}_{pw_{i,d_N}}(Y_{j,d_j})$ , if $(X_{j,d_N} \in E(\mathbb{F}_q))$ :
⑨	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_j}\}$	$\text{sid}_{i,j,d_2} = \{d_2, Y_{i,d_2}, d_j, Y_{j,d_j}\}$		$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_j}\}$
⑩	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_j}, x_{i,d_1} \cdot X_{j,d_j} \bmod q)$	$\text{sk}_{i,j,d_2} \leftarrow H(d_2, d_j, X_{i,d_2}, X_{j,d_j}, x_{i,d_2} \cdot X_{j,d_j} \bmod q)$		$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_j}, x_{i,d_N} \cdot X_{j,d_j} \bmod q)$
⑪	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
⑫	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
⑬		For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :		
⑭	if $(Y_{i,d_1} \in \text{sid}_{i,j,d_j})$ : $r_{i,j,d_1} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_j})$	if $(Y_{i,d_2} \in \text{sid}_{i,j,d_j})$ : $r_{i,j,d_2} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_j})$		if $(Y_{i,d_N} \in \text{sid}_{i,j,d_j})$ : $r_{i,j,d_N} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_j})$
⑮	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$

Generate session id using received ID and pks

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
<b>Step 1: Evidence Extraction</b>				
1	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$			$\dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
2	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \lfloor \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \rfloor$			$\dots pw_{c_N,d_N}\} = \lfloor \{i_{1,d_1} \dots i_{c_N,d_N}\} / W \rfloor$
3	Determine the public parameters, two p			Curve, and $P \in E(\mathbb{F}_q)$ is its generator.
4	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$	Choose random $\{x_{1,d_2} \dots x_{c_2,d_2}\} \xleftarrow{\$} \mathbb{Z}_p$		Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
5	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \bmod q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$	$X_{i,d_2} \leftarrow x_{i,d_2} \cdot P \bmod q, Y_{i,d_2} \leftarrow \text{Enc}_{pw_{i,d_2}}(X_{i,d_2})$		$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \bmod q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
6	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$	Broadcast $(d_2, Y_{i,d_2})$ , where $i \in \{1, \dots, c_2\}$		Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
7		For every received message $(d_j, Y_{j,d_j})$ , where $j \in \{1, \dots, N\}$ :		
8	$X_{j,d_1} \leftarrow \text{Dec}_{pw_{i,d_1}}(Y_{j,d_j})$ , if $(X_{j,d_1} \in E(\mathbb{F}_q))$ :	$X_{j,d_2} \leftarrow \text{Dec}_{pw_{i,d_2}}(Y_{j,d_j})$ , if $(X_{j,d_2} \in E(\mathbb{F}_q))$ :		$X_{j,d_N} \leftarrow \text{Dec}_{pw_{i,d_N}}(Y_{j,d_j})$ , if $(X_{j,d_N} \in E(\mathbb{F}_q))$ :
9	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_j}\}$	$\text{sid}_{i,j,d_2} = \{d_2, Y_{i,d_2}, d_j, Y_{j,d_j}\}$		$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_j}\}$
10	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_j}, x_{i,d_1} \cdot X_{j,d_j} \bmod q)$	$\text{sk}_{i,j,d_2} \leftarrow H(d_2, d_j, X_{i,d_2}, X_{j,d_j}, x_{i,d_2} \cdot X_{j,d_j} \bmod q)$		$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_j}, x_{i,d_N} \cdot X_{j,d_j} \bmod q)$
11	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
12	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
13		For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :		
14	if $(Y_{i,d_1} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_1} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_j})$	if $(Y_{i,d_2} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_2} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_j})$		if $(Y_{i,d_N} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_N} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_j})$
15	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$

Derive intermediate two-party ECDH keys

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
	<b>Step 1: Evidence Extraction</b>			
1	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$			$\{i_{1,d_N} \dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
2	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \{i_{1,d_1} \dots i_{c_1,d_1}\} / W$			
3	Determine the public parameters, two p			
4	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \leftarrow \mathbb{Z}_p$	Choose random $\{x_{1,d_2} \dots x_{c_2,d_2}\} \leftarrow \mathbb{Z}_p$		Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \leftarrow \mathbb{Z}_p$
5	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \text{ mod } q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$	$X_{i,d_2} \leftarrow x_{i,d_2} \cdot P \text{ mod } q, Y_{i,d_2} \leftarrow \text{Enc}_{pw_{i,d_2}}(X_{i,d_2})$		$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \text{ mod } q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
6	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$	Broadcast $(d_2, Y_{i,d_2})$ , where $i \in \{1, \dots, c_2\}$		Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
7		For every received message $(d_j, Y_{j,d_2})$ , where $j \in \{1, \dots, N\}$ :		
8	$X_{j,d_1} \leftarrow \text{Dec}_{pw_{i,d_1}}(Y_{j,d_1})$ , if $(X_{j,d_1} \in E(\mathbb{F}_q))$ :	$X_{j,d_2} \leftarrow \text{Dec}_{pw_{i,d_2}}(Y_{j,d_2})$ , if $(X_{j,d_2} \in E(\mathbb{F}_q))$ :		$X_{j,d_N} \leftarrow \text{Dec}_{pw_{i,d_N}}(Y_{j,d_N})$ , if $(X_{j,d_N} \in E(\mathbb{F}_q))$ :
9	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_1}\}$	$\text{sid}_{i,j,d_2} = \{d_2, Y_{i,d_2}, d_j, Y_{j,d_2}\}$		$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_N}\}$
10	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_1}, x_{i,d_1} \cdot X_{j,d_1} \text{ mod } q)$	$\text{sk}_{i,j,d_2} \leftarrow H(d_2, d_j, X_{i,d_2}, X_{j,d_2}, x_{i,d_2} \cdot X_{j,d_2} \text{ mod } q)$		$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_N}, x_{i,d_N} \cdot X_{j,d_N} \text{ mod } q)$
11	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
12	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
13		For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :		
14	if $(Y_{i,d_1} \in \text{sid}_{i,j,d_1}) : r_{i,j,d_1} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_1})$	if $(Y_{i,d_2} \in \text{sid}_{i,j,d_2}) : r_{i,j,d_2} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_2})$		if $(Y_{i,d_N} \in \text{sid}_{i,j,d_N}) : r_{i,j,d_N} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_N})$
15	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$

generate random values for each event type, encrypts them using the intermediate keys, and broadcasts along with its ID and the session ID

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
	<b>Step 1: Evidence Extraction</b>			
1	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$			$\{i_{1,d_N} \dots i_{c_N,d_N}\} = \text{CONTEXT\_EXTRACTION}(d_N)$
2	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \{ \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \}$			
3	Determine the public parameters, two p			
4	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$	Choose random $\{x_{1,d_2} \dots x_{c_2,d_2}\} \xleftarrow{\$} \mathbb{Z}_p$		Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
5	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \bmod q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$	$X_{i,d_2} \leftarrow x_{i,d_2} \cdot P \bmod q, Y_{i,d_2} \leftarrow \text{Enc}_{pw_{i,d_2}}(X_{i,d_2})$		$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \bmod q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
6	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$	Broadcast $(d_2, Y_{i,d_2})$ , where $i \in \{1, \dots, c_2\}$		Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
7		For every received message $(d_j, Y_{j,d_2})$ , where $j \in \{1, \dots, N\}$ :		
8	$X_{j,d_1} \leftarrow \text{Dec}_{pw_{i,d_1}}(Y_{j,d_1})$ , if $(X_{j,d_1} \in E(\mathbb{F}_q))$ :	$X_{j,d_2} \leftarrow \text{Dec}_{pw_{i,d_2}}(Y_{j,d_2})$ , if $(X_{j,d_2} \in E(\mathbb{F}_q))$ :		$X_{j,d_N} \leftarrow \text{Dec}_{pw_{i,d_N}}(Y_{j,d_N})$ , if $(X_{j,d_N} \in E(\mathbb{F}_q))$ :
9	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_1}\}$	$\text{sid}_{i,j,d_2} = \{d_2, Y_{i,d_2}, d_j, Y_{j,d_2}\}$		$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_N}\}$
10	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_1}, x_{i,d_1} \cdot X_{j,d_1} \bmod q)$	$\text{sk}_{i,j,d_2} \leftarrow H(d_2, d_j, X_{i,d_2}, X_{j,d_2}, x_{i,d_2} \cdot X_{j,d_2} \bmod q)$		$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_N}, x_{i,d_N} \cdot X_{j,d_N} \bmod q)$
11	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
12	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
13		For every received message $(d_j, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :		
14	if $(Y_{i,d_1} \in \text{sid}_{i,j,d_1}) : r_{i,j,d_1} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_1})$	if $(Y_{i,d_2} \in \text{sid}_{i,j,d_2}) : r_{i,j,d_2} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_2})$		if $(Y_{i,d_N} \in \text{sid}_{i,j,d_N}) : r_{i,j,d_N} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_N})$
15	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_1})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_2})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_N})$

Check if its ID is in the session ID of the received message. If yes, decrypt with intermediate key to get random number of other device

# New Key Establishment Protocol



	Device 1 ( $d_1$ )	Device 2 ( $d_2$ )	...	Device N ( $d_N$ )
<b>Step 1: Evidence Extraction</b>				
1	$\{i_{1,d_1} \dots i_{c_1,d_1}\} = \text{CONTEXT\_EXTRACTION}(d_1)$	<b>Add random values to derive group keys</b>		
2	$\{pw_{1,d_1} \dots pw_{c_1,d_1}\} = \lfloor \{i_{1,d_1} \dots i_{c_1,d_1}\} / W \rfloor$	$\{pw_{1,d_2} \dots pw_{c_2,d_2}\} = \lfloor \{i_{1,d_2} \dots i_{c_2,d_2}\} / W \rfloor$		$\{pw_{1,d_N} \dots pw_{c_N,d_N}\} = \lfloor \{i_{1,d_N} \dots i_{c_N,d_N}\} / W \rfloor$
<b>Step 3: Partitioned GPAKE</b>				
3	Determine the public parameters, two primes $p$ and $q$ , a finite field $\mathbb{F}_q$ and a group $\mathbb{Z}_p$ . $E(\mathbb{F}_q)$ is an elliptic curve, and $P \in E(\mathbb{F}_q)$ is its generator.			
4	Choose random $\{x_{1,d_1} \dots x_{c_1,d_1}\} \xleftarrow{\$} \mathbb{Z}_p$	Choose random $\{x_{1,d_2} \dots x_{c_2,d_2}\} \xleftarrow{\$} \mathbb{Z}_p$		Choose random $\{x_{1,d_N} \dots x_{c_N,d_N}\} \xleftarrow{\$} \mathbb{Z}_p$
5	$X_{i,d_1} \leftarrow x_{i,d_1} \cdot P \text{ mod } q, Y_{i,d_1} \leftarrow \text{Enc}_{pw_{i,d_1}}(X_{i,d_1})$	$X_{i,d_2} \leftarrow x_{i,d_2} \cdot P \text{ mod } q, Y_{i,d_2} \leftarrow \text{Enc}_{pw_{i,d_2}}(X_{i,d_2})$		$X_{i,d_N} \leftarrow x_{i,d_N} \cdot P \text{ mod } q, Y_{i,d_N} \leftarrow \text{Enc}_{pw_{i,d_N}}(X_{i,d_N})$
6	Broadcast $(d_1, Y_{i,d_1})$ , where $i \in \{1, \dots, c_1\}$	Broadcast $(d_2, Y_{i,d_2})$ , where $i \in \{1, \dots, c_2\}$		Broadcast $(d_N, Y_{i,d_N})$ , where $i \in \{1, \dots, c_N\}$
7		For every received message $(d_j, Y_{j,d_j})$ , where $j \in \{1, \dots, N\}$ :		
8	$X_{j,d_1} \leftarrow \text{Dec}_{pw_{i,d_1}}(Y_{j,d_1})$ , if $(X_{j,d_1} \in E(\mathbb{F}_q))$ :	$X_{j,d_2} \leftarrow \text{Dec}_{pw_{i,d_2}}(Y_{j,d_2})$ , if $(X_{j,d_2} \in E(\mathbb{F}_q))$ :		$X_{j,d_N} \leftarrow \text{Dec}_{pw_{i,d_N}}(Y_{j,d_N})$ , if $(X_{j,d_N} \in E(\mathbb{F}_q))$ :
9	$\text{sid}_{i,j,d_1} = \{d_1, Y_{i,d_1}, d_j, Y_{j,d_j}\}$	$\text{sid}_{i,j,d_2} = \{d_2, Y_{i,d_2}, d_j, Y_{j,d_j}\}$		$\text{sid}_{i,j,d_N} = \{d_N, Y_{i,d_N}, d_j, Y_{j,d_j}\}$
10	$\text{sk}_{i,j,d_1} \leftarrow H(d_1, d_j, X_{i,d_1}, X_{j,d_j}, x_{i,d_1} \cdot X_{j,d_j} \text{ mod } q)$	$\text{sk}_{i,j,d_2} \leftarrow H(d_2, d_j, X_{i,d_2}, X_{j,d_j}, x_{i,d_2} \cdot X_{j,d_j} \text{ mod } q)$		$\text{sk}_{i,j,d_N} \leftarrow H(d_N, d_j, X_{i,d_N}, X_{j,d_j}, x_{i,d_N} \cdot X_{j,d_j} \text{ mod } q)$
11	$r_{i,d_1} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_1} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_1}}(r_{i,d_1})$	$r_{i,d_2} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_2} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_2}}(r_{i,d_2})$		$r_{i,d_N} \xleftarrow{\$} \mathbb{Z}_p, \alpha_{i,j,d_N} \leftarrow \text{Enc}_{\text{sk}_{i,j,d_N}}(r_{i,d_N})$
12	Broadcast $(d_1, \text{sid}_{i,j,d_1}, \alpha_{i,j,d_1})$	Broadcast $(d_2, \text{sid}_{i,j,d_2}, \alpha_{i,j,d_2})$		Broadcast $(d_N, \text{sid}_{i,j,d_N}, \alpha_{i,j,d_N})$
13	For every received message $(d_j, \text{sid}_{i,j,d_j}, \alpha_{i,j,d_j})$ , where $i \in \{1, \dots, c\}$ and $j \in \{1, \dots, N\}$ :			
14	if $(Y_{i,d_1} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_1} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_1}}(\alpha_{i,j,d_j})$	if $(Y_{i,d_2} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_2} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_2}}(\alpha_{i,j,d_j})$		if $(Y_{i,d_N} \in \text{sid}_{i,j,d_j}) : r_{i,j,d_N} \leftarrow \text{Dec}_{\text{sk}_{i,j,d_N}}(\alpha_{i,j,d_j})$
15	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$	$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$		$\text{key}_i \leftarrow \sum_{j=1}^t (r_{i,j,d_j})$