Start

if (Log Counter == 0)
    do Enclave-initialisation
else
    skip

Idle/collection mode

data collection

training finished

receive start_training() command from insurance

Training

DP-GBDT Enclave

1a. Inc()

1b. $c_{log} \coloneqq$ Read()

2. randomly choose master seed s_m

3. $Log_{init} \coloneqq$ seal({"Log", id_{LC}, c_{log}, s_m})

Log Counter (id_{LC})

[val = 0]
[val = 1]

Harddisk

Log_{init}

DP-GBDT Enclave

1. Attestation

1b. Establish secure session

2. Send encrypted questionnaire $:= Q_i$

3. $decrypt(Q_i) \rightarrow data_i$

4a. $Inc()$

4b. $c_i := Read()$

5. ensure harddisk contains $D_1 \dots D_{i-1}$ and they all contain master seed s_m from Log_{init}

6. $D_i := seal(\{data_i, s_m, c_i\})$

Questionnaire Counter

[val = i-1]
[val = i]

Harddisk

D_1 Log_{init}
:
D_{i-1}
D_i

DP-GBDT Enclave

1. train(n = 500)

2a. ensure n is valid training size

2b. $c_q \coloneqq$ Read()

2c. ensure $c_q \geq n$

3a. load* Log_{init} and $D_1...D_{500}$ into enclave

4. (prng-)derive s_1 from s_m

5a. Inc()

5b. $c_{log} \coloneqq$ Read()

6. $Log_1 \coloneqq$ seal({"Log", id_{LC}, c_{log}, 1, 500, s_1})

7. start_training($D_1..D_{500}$, s_1)

8. output model

Harddisk

D_1 Log_{init}
: Log_1
D_{500}
:
D_{505}

Questionnaire Counter
[val = 505]

Log Counter (id_{LC})
[val = 1]
[val = 2]

DP-GBDT Enclave

ZURICH

1. train(n = 700)

2a. ensure n is valid training size

2b. c_q ≔ Read()

2c. ensure $c_q \geq n$

Questionnaire Counter

[val = 724]

Log Counter (id_{LC})

[val = 3]

3. c_{log} ≔ Read() = 7

Harddisk

D_1 Log_{init}
: Log_1
D_{500} Log_2
:
D_{724}

4a. load* Log_3 from disk

4b. load* $D_{501}..D_{700}$ from disk

5. start_training($D_{501}..D_{700}$, s_3)

6. output model

Master Enclave

sk_M/pk_M

$E_1 - pk_{E1}$
$E_2 - pk_{E2}$
$E_3 - pk_{E3}$

$\{Q_i\}_{pk_M}$

$\{Q_i\}_{pk_{Ej}}$

DP-GBDT Enclave

sk_{E1}/p

DP-GBDT Enc

sk_{E2}/p

DP-GBDT Enclave E_3

sk_{E3}/pk_{E3}

```
cpp_gbdt                    python_gbdt

hardened_gbdt      enclave_gbdt

                   hardened_
                   enclave_gbdt
```