1 Encryption procedure

Suppose actuators and sensors to be trustworthy, therefore we can use a symmetric key algorithm, as it is much faster than an asymmetric one.

To encrypt the remote fault detection system, we will present a SWHE symmetric key scheme inspired by the one of van Dijk et al. [**?**]

The plaintext domain $\mathcal{M}$ of the scheme only allows for encryption of M natural numbers

$$\mathcal{M} \in \mathbb{N} \cap [0, \dots, M-1]$$

The key generation algorithm KeyGen will generate two large enough private not necessarily equal prime numbers (ideally at least 1024 bits each)

$$p, q \leftarrow\$ \text{ KeyGen}(1^\lambda) \tag{1}$$

The length constraint is place to ensure that the factorization of $N = pq$ is computationally difficult: being a *semiprime number* (the result of a product of two prime numbers), its factorization can be obtained only through an enumeration attack if p and q are unknown and large enough.

The problem with this scheme, like all SWHE schemes, is that it allows for a limited number of multiplications Ω: after this value has been determined for the specific implementation and the largest possible integer value M is known, for the scheme to allow for correct decryption, p, q shall be picked as follows

$$\begin{aligned} p &\in [2^{\mu-1}, 2^\mu] \\ q &\in [2^{\eta-1}, 2^\eta] \end{aligned} \qquad \left(\mu \approx \frac{3\Omega}{2\log_2(M)},\ \eta \approx \frac{3\Omega\mu}{2-\mu}\right) \tag{2}$$

Then, the symmetric key k would be p.

The parameter N is made public, as it is needed to encrypt and decrypt: it encases p, q but hidden behind the computational complexity of the factorization of large primes.

The operations on data we can perform are:

$$\begin{aligned} &\text{Enc}(m, p) = (m + wp) \bmod N \qquad w \leftarrow\$ [1, q-1] \\ &\text{Dec}(c, p) = c \bmod p = (m + wp) \bmod p = m \end{aligned} \tag{3}$$

The random value w represents random noise that added to the message to mask the original data.

1.1 Homomorphic operations

Let two cyphertexts c, c' obtained from the encryption of two different messages m, m' from the aforementioned SWHE scheme

$$\begin{aligned} c &= \text{Enc}(m, p) = m + wp \\ c' &= \text{Enc}(m', p) = m' + w'p \end{aligned} \qquad w, w' \leftarrow\$ [1, q-1] \tag{4}$$

The scheme admits two homomorphic operations

$$\begin{aligned} c \oplus c' &= (c + c') \bmod N = (m + m') + (w + w')p \bmod N \\ c \otimes c' &= (cc') \bmod N = (mm') + (wm' + w'm + ww'p)p \bmod N \end{aligned} \tag{5}$$

By decrypting, in fact, we obtain

$$\begin{aligned} \mathrm{Dec}(c \oplus c') &= (m + m') + (w + w')p \bmod p = m + m' \qquad & (m + m' < p) \\ \mathrm{Dec}(c \otimes c') &= (mm') + (wm' + w'm + ww'p)p \bmod p = mm' \qquad & (mm' < p) \end{aligned} \tag{6}$$

1.2 Mapping function

The mentioned scheme is designed such that it only allows for the encryption of natural numbers. Therefore, we require a mapping function from real (and possibly negative) numbers to natural numbers to correctly process signals from cyber-physical systems.

Say we need to encrypt the real value ξ such that

$$\xi \in [-\xi_{\max}, \xi_{\max}], \quad \xi_{\max} \in \mathbb{N}$$

To achieve the above, let us define the following mapping function $\Gamma(\xi, \xi_{\max}) : \mathbb{R} \rightarrow \mathcal{M}$

$$\Gamma(\xi, \xi_{\max}) = \begin{cases} round(\xi) & \xi \geq 0 \\ 2\xi_{\max} + 1 + round(\xi) & \xi < 0 \end{cases} \tag{7}$$

Where $round(\xi)$ rounds ξ to the nearest integer with quantization error up to 0.5.

This mapping requires the presence of at least $2\xi_{\max} + 1$ values in the plaintext space, i.e. $|\mathcal{M}| \geq 2\xi_{\max} + 1$. In our encryption scheme we are limited by the decryption function to $|\mathcal{M}| = p$, thus $\xi_{\max} \geq \frac{p-1}{2}$

$$\xi \in \left[-\frac{p-1}{2}, \frac{p-1}{2}\right] \tag{8}$$

The following figure shows the mapping of ξ with $\xi_{\max} = 5$, thus $|\mathcal{M}| \geq 11$

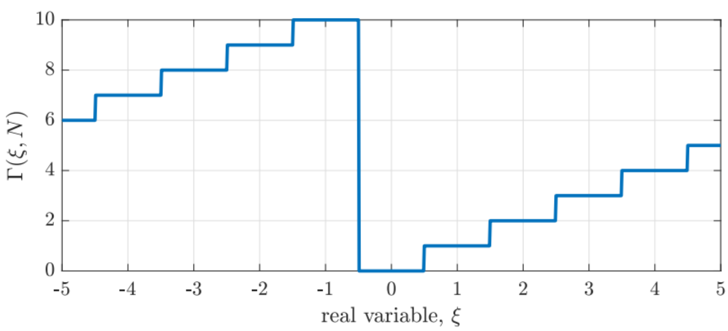

Figure 1: Conversion from real value to positive integer

The quantization error can be reduced significantly with the introduction of a gain γ, which simply multiplies ξ before the mapping

$$\gamma \longrightarrow \Gamma(\gamma\xi, \xi_{\text{max}}) = \bar{\xi} \tag{9}$$

This gain ensures the precision of the mapping up to $\log_{10}\gamma$ decimal points, but requires $\gamma\xi$ more values in the plaintext domain $|\mathcal{M}|$. In other words, γ increases the granularity but decreases the reach.

$$\xi = \Gamma^{-1}(\bar{\xi}, \gamma, \xi_{\text{max}}) = \begin{cases} \bar{\xi}\gamma^{-1} & \bar{\xi} \geq \xi_{\text{max}} + 1 \\ (\bar{\xi} - 2\xi_{\text{max}} - 1)\gamma^{-1} & \bar{\xi} < \xi_{\text{max}} + 1 \end{cases} \tag{10}$$

We will also require a reversed mapping $\Gamma^{-1}(\bar{\xi}, \gamma, \xi_{\text{max}})$ to obtain the original ξ after decryption. If scaling was used, after decryption the result will be re-scaled.

1.3 Multiplication depth

An unpleasant side-effect of SWHE schemes is that as we keep working in the encrypted domain, the gain γ accumulates after every subsequent multiplication: the number of times we do that is named multiplication depth ω.

Definition 1.1 (Multiplication depth ω)**.** The number of consecutive homomorphic multiplications performed on fresh cyphertexts.

Definition 1.2 (Fresh cyphertext)**.** A cyphertext obtained directly after encryption of a plaintext, always has $\omega = 1$.

The multiplication depth ω is unfortunately finite: there is an upper bound after which random noise makes the decryption of the single multiplicative terms impossible.

In our case, $\omega \leq \Omega$ should be respected for decryption to succeed

$$\Omega \in \mathbb{N} : \gamma^{\Omega}\xi < p \tag{11}$$

This limitation is one of the main obstacles when it comes to employing SWHE schemes to protect dynamical systems' data exchanges.

Important note: during re-scaling of a value, the result will be scaled by $\gamma^{-\omega}$, which is a multiplicative factor: this implies that in an homomorphic addition, every term must have the same exact ω

$$\left.\begin{aligned} \xi_1 \to \gamma\xi_1 \\ \xi_2 \to \gamma\xi_2 \end{aligned}\right\} \longrightarrow \text{Dec}(c_1 \otimes c_2) = \gamma^2\xi_1\xi_2 \longrightarrow \xi = \Gamma^{-1}(\gamma^2\xi_1\xi_2, \gamma^2, \xi_{\text{max}}) \tag{12}$$

1.4 Encrypted scenario

Let's see the scenario: the encrypted cyber-physical system (**??**) is equipped with observer (**??**) and the controller is shown in detail in the following figure.

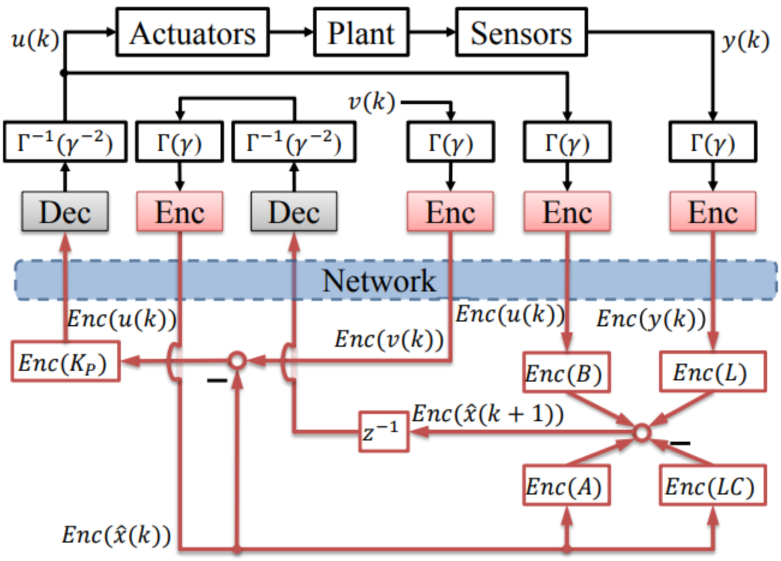

Figure 2: Encrypted CPS with observer and controller

For simplicity, assume it being an observer-based state feedback controller with feedback gain K_p and setpoint (i.e. reference value) $v(k)$

$$u(k) = K_p(v(k) - \hat{x}(k)) \tag{13}$$

Also assume unitary weighting matrix $W = 1$, turning (**??**) into

$$\mathcal{D} : \begin{cases} \hat{x}(k+1) = A\hat{x}(k) + Bu(k) + Lr(k) \\ \hat{y}(k) = C\hat{x}(k) \\ r(k) = y(k) - \hat{y}(k) \end{cases} \tag{14}$$

This way, $\hat{x}(k+1) = A\hat{x}(k) + Bu(k) + Ly(k) - LC\hat{x}(k)$.

We can see that every signal (v, u, y) and every matrix (A, B, C, L, K_p) is scaled by γ, mapped and then encrypted, thus will all have $\omega = 1$. The mapping operation is represented by $\Gamma(\gamma)$ for simplicity.

The encrypted state estimation $\text{Enc}(\hat{x}(k))$ is obtained through an homomorphic addition involving all addends with the same multiplication depth of $\omega = 2$, which correctly satisfies the aforementioned property

$$\begin{aligned} \text{Enc}(\hat{x}(k+1)) = {} & [\text{Enc}(A) \otimes \text{Enc}(\hat{x}(k))] \oplus [\text{Enc}(B) \otimes \text{Enc}(u(k))] \\ & \oplus [\text{Enc}(L) \otimes \text{Enc}(y(k))] \oplus [\text{Enc}(-LC) \otimes \text{Enc}(\hat{x}(k))]. \end{aligned}$$

To avoid accumulation of ω in the encrypted state estimation signal, we send it through the network to the plant side, decrypt it and encrypt it again such that it has $\omega = 1$: this operation is often referred as *refreshing*.

The encrypted control signal would be

$$\text{Enc}(u(k)) = \text{Enc}(K_p) \otimes [\text{Enc}(v(k)) \oplus \text{Enc}(-\hat{x}(k))] \tag{15}$$

Note how above encrypted control input signal $\text{Enc}(u(k)))$ and the encrypted state estimation $\text{Enc}(\hat{x}(k+1))$ are both involved in an homomorphic multiplication, thus they have $\omega = 2$, thus during re-scaling their decryption will be scaled by γ^{-2}.

1.5 Secret residual evaluation

The detection of faults is based on the residual signal r generated by the observer-based fault detector (**??**) previously presented in (**??**).

The previously mentioned detector would evaluate $||r||_E$ and compare it with a treshold τ, however it involves the use of the square root, an operation that is not defined in our cypher scheme.

To address this problem, $||r||_E$ is still evaluated in the plaintext domain, but then it is squared, encrypted and homomorphically summed with the opposite (i.e. subtracted) of the threshold τ modified in the same way. The result is then decrypted and finally evaluated by checking its sign

$$\begin{aligned} &\text{Dec}(\text{Enc}(||r||_E^2) \oplus \text{Enc}(-\tau^2)) \leq 0 \qquad \text{normal behavior} \\ &\text{Dec}(\text{Enc}(||r||_E^2) \oplus \text{Enc}(-\tau^2)) > 0 \qquad \text{abnormal behavior} \end{aligned} \tag{16}$$

This allows the defender of the system to store the value τ as cyphertext, this is significant as its plaintext counterpart contains useful information for the attacker to mask a potential cyberattack.