Location Dependent Cryptosystem

By: Kunal Mukherjee, Computer Engineering

Project Sponsor: **Mr. Mike Ciholas**
Academic Advisor: **Dr. Donald Roberts**
Design Advisor: **Mr. Justin Bennett**
Software Advisor: **Mr. Tim DeBaillie**

Abstract

Certain research dependent industries need new security measures to protect their intellectual property from corporate or international espionage. For this reason, a location dependent cryptosystem which maps the key to the approved location was necessary. The cryptosystem will allow the user to decrypt, only if the user is at an approved location which is predetermined by the sender. This system will also function without the receiver's assistance or knowledge of the password**.**

Cryptosystem Goal

- The cryptosystem's transmitter steps :
  - A. Take the password
  - B. Use SHA256, generates fixed size 256-bit (32-byte) hash
  - C. Use the hash to encrypt the data file
  - D. Encode the bytes of hash into time stamp
  - E. Transmit the encrypted data at the correct time stamp
- The cryptosystem's reception steps :
  - A. Receives all the encrypted data
  - B. Use the reception time stamp to generate the corresponding hash
  - C. Use the hash to decrypt the encrypted data

Description of Constants and Variables

- Network ticks (NT) is the clock is being run by the Ciholas server, NetApp
- UWB waves propagate in air at the speed of light in vacuum, C , which is 299,792,458 meters per second
- **Kf** is the conversion factor that changes time (sec) to network ticks (NT)
  - A. 1 sec = **97500 * 65536 NT ticks**
- Initial Network cadence is the amount of time the network needs to start
  - **A. Tnet** = multiple of **97500 * 65536**
  - **B. TstartOffset** = **5 msec** or **5 msec * Kf NT ticks**
- Packet arrival window is the time frame that is allotted for the transfer of each byte of the hash
  - **A. TwinOffset = 2.5 msec** or **2.5 msec * Kf NT ticks** wide
  - B. provides ample time that the device needs after it receives a packet, so that it can process the packet and go back to listening mode again
- The approved space, sphere, where the time stamp can be utilized to generate the byte of hash
  - **A. Tslot for a approved enclosed space of diameter d meter = d / C**
    - **I. Tslot for 2 m = 6.67 ns**
- The amount of time packets needs to get from anchor, located at ***d*** distance to the approved location, is TdA
  - **A. TdA** = ***d*** **/ C sec** or ***d*** **/ C * Kf NT ticks**
- Slot Number or Slot# is the slot in time that corresponds to the value of the byte of SHA256 hash
  - **A. SHA value** of **5D** (hex) or **93** (dec) would be allotted Slot# **93**

Conclusion

The location dependent encryption system introduced accurately connects the passwords of an encrypted data to the location. Only user is at the authorized location is able to decrypt the file. The system is still vulnerable to various attacks. For example, the use of three listening anchors can be used to easily limit the search space of the approved location as well as guessing the transmission anchor position. Therefore, future work needs to be done in developing OTP (one time password) that is time synchronized.

Transmission Equation:

$$\mathrm{Ttx}(0) = \mathrm{Tnet} + \mathrm{TstartOffset}$$

$$\mathrm{Ttx}(n) = \mathrm{Ttx}(n-1) + \mathrm{TdA}(n-1) + \mathrm{TwinOffset} - \mathrm{TdA}(n) + ((\mathrm{Slot\#} + 0.5) * \mathrm{Tslot})$$

Receiving Equation:

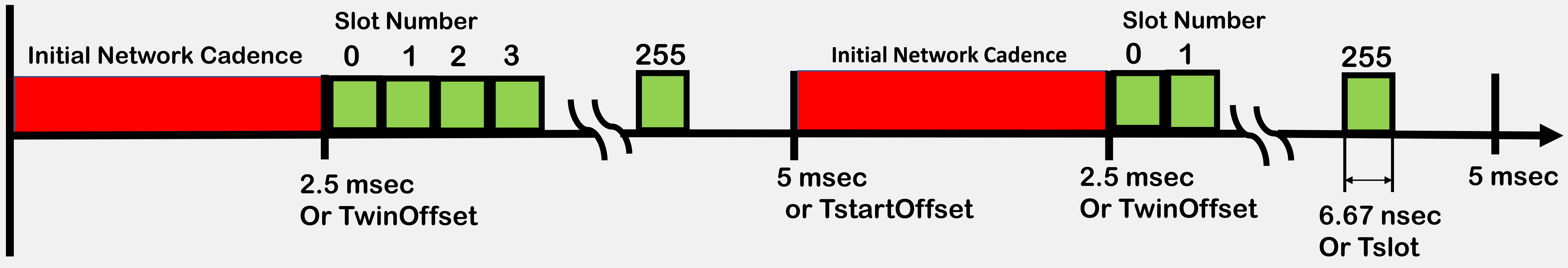
$$\mathrm{SlotValue}(n) = (\mathrm{Trx}(n) - \mathrm{Trx}(n-1) - \mathrm{TwinOffset}) / \mathrm{Tslot}$$

Security Feature

- The transmitter uses the transmission equation to generate the time stamps required to encode a byte of the hash
  - A. For example, let **5D** be a specific SHA byte that needs to be transmitted corresponding to **slot number 93**
    - 1) the transmitter will send a packet at ***x*** **NT *ticks*** which can be used to calculate the **slot number *93***
    - 2) transmitter will **randomly** choose an anchor, A, which is at a known distance ***d distance or TdA NT ticks*** away from the approved location
      - I. the transmission time for the packet would be calculated so that the packet is transmitted at ***x – TdA* NT *ticks***
- The receiver who is at the approved location which is at ***d distance*** or ***t* NT *ticks*** from the anchor
  - A. Will receive the packet at ***x – TdA + TdA* NT *ticks*** or ***x* NT *ticks***
  - ***B. x* NT *ticks*** would generate the slot value of ***93***, which is **same** as the intended slot value
  - C. Receiver will be **able to recreate** the specific byte of hash, **5D**
- Eavesdropper who is **not** at the approved location which is at ***d' distance*** or ***t'* NT *ticks*** from the transmission anchor
  - A. Will receive the packet at ***x – TdA + t'* NT *ticks***
  - ***B. x – TdA + t'* NT *ticks*** would generate the slot value of ***s'***, which is **not same** as the intended slot value
  - C. Thus, **preventing** the eavesdropper from recreating the **correct hash** corresponding to the encryption key

The security is established as without knowing the **randomly chosen anchor's location** or being at the **approved location, dA meter from the randomly chosen anchor,** the slot number could not be calculated accurately.

Time Frame



Initial Network Cadence
Slot Number
0
1
2
3
255
2.5 msec
Or TwinOffset
Initial Network Cadence
Slot Number
0
1
255
5 msec
or TstartOffset
2.5 msec
Or TwinOffset
6.67 nsec
Or Tslot
5 msec