Improving PHY-Layer Security using Probabilistic Symbol Extension

Garrett Vanhoy[1], Tamal Bose[1]

[1]University of Arizona Electrical and Computer Engineering Dept.

THE UNIVERSITY OF ARIZONA

Agenda

- PHY-layer Security
- Probabilistic Symbol Extension
- Analysis and Results

PHY-Layer Security

THE UNIVERSITY OF ARIZONA

Physical Layer Security

- Wireless networks are naturally vulnerable to attacks
- Encryption can prevent many basic attacks, but is not a complete solution.
- Side-channel information (SCI) can still be exploited to perform crippling attacks.

Side-Channel Information

- Side-channel information (SCI) is anything other than the original data or the encrypted data itself.

- Each layer of the OSI stack exposes some kind SCI, but the MAC and PHY layers are especially vulnerable in wireless networks

SCI Examples

- **Physical layer SCI**
  - **signal strength, bandwidth, modulation, channel coding, etc.**

- **Frame size, frame interarrival times, and packet direction are critical**
  - **Can classify: application being used**
  - **Specific search query**

- **Looking at these is generally called *traffic analysis.***

Friendly Jamming

- Technique: Conceal the signal
  - Reduce the effective SNR at the eavesdropper by jamming in the unused signal space (space, time, or frequency)

- Limitations
  - Where is the eavesdropper? They are silent!
  - With enough antennas or participating nodes, the likelihood this works decreases significantly.

Full-Frame Encryption

- Technique: <u>Conceal the meaning</u>
  - Encrypt everything, including PHY headers.
  - <u>Ideally:</u> the signal becomes a mostly meaningless mess of bits to the eavesdropper.

- Limitations
  - The number of bits in each frame is not hidden and since many control packets have a known length, it is still possible to perform traffic analysis.

Obfuscation

- Technique: <u>Confuse the jammer</u>
  - Make observable parameters not as they seem.
  - Example: modulation and coding scheme, MAC address, etc.

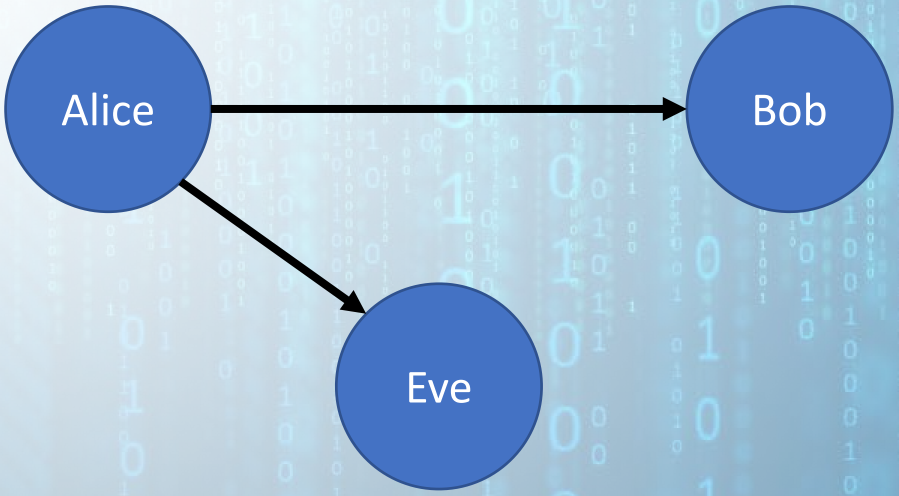
- Existing techniques
  - Padding frames with dummy data
  - Modulation obfuscation maps lower order modulations onto higher ones

Probabilistic Symbol Extension

THE UNIVERSITY OF ARIZONA

Probabilistic Symbol Extension

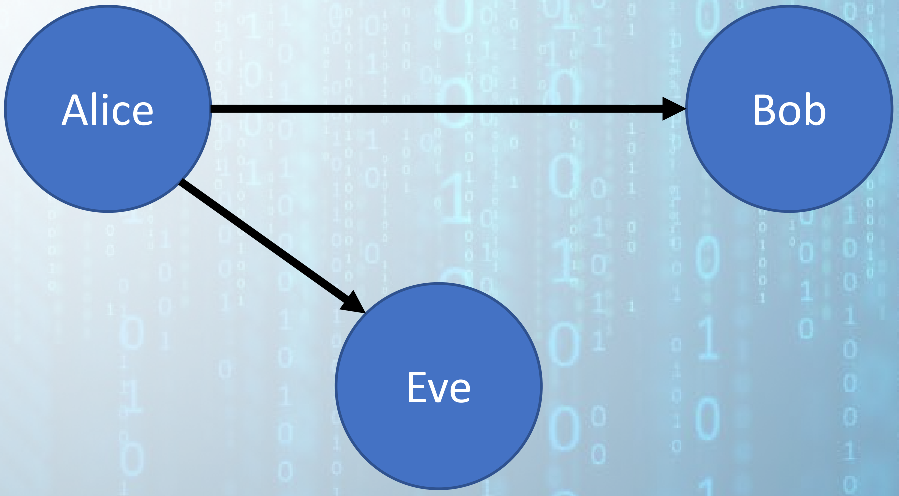
- Alice and Bob have a shared secret key with which to generate a shared secret sequence
- Using the secret sequence, Alice extends some symbols.



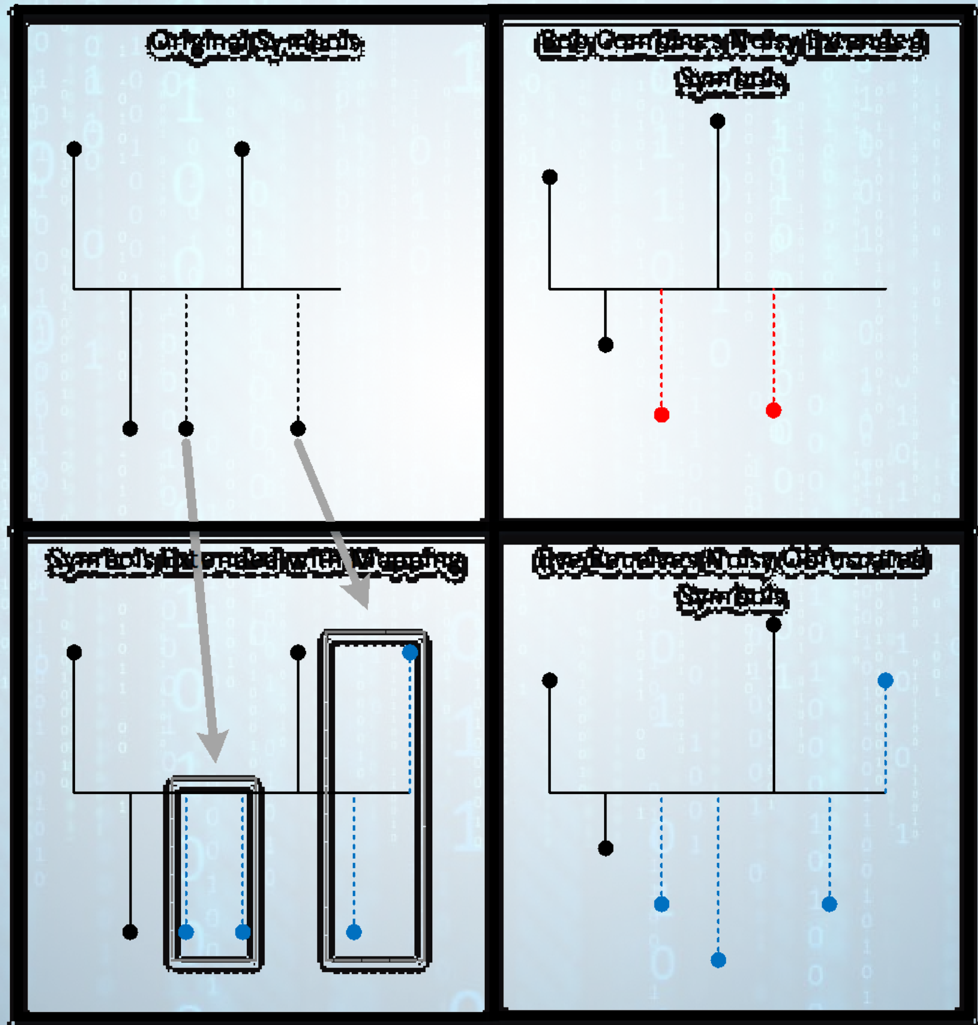
Probabilistic Symbol Extension

- Bob knows which symbols are extended and can compensate

- Eve sees more symbols and thus sees a larger frame size



Probabilistic Symbol Extension



Probabilistic Symbol Extension

- To extend BPSK symbols in a way that Eve cannot detect this extension we have a few steps:

1. Choose symbols to extend to R symbols.
2. Map them to a random set of R symbols.
3. Transform them by the original symbol.

Probabilistic Symbol Extension

- Given $I = [i_1, i_2, \dots, i_{L_I}]$, a shared secret sequence of integers, we can use every odd integer to choose which symbols to extend.

- We can design this mapping so that a portion of the symbols are extended.

Probabilistic Symbol Extension

- If we want roughly half of the symbols to be extended to two symbols $(P_2 \approx .5)$, we have a simple rule:
  - If the element is less than half of the maximum value, extend it.
  - Leave it alone otherwise.

Probabilistic Symbol Extension

- For each symbol that is extended, use the even values of $I_e = \left[i_2, i_4, \dots, i_{L_I/2}\right]$ to determine which set of two symbols T^2 should be used.

- Given i_2 is an integer, we can split its range of values into four equally sized regions and map T^2 onto any one of:
  - [1, 1], [1, -1], [-1, 1], or [-1, -1]

Probabilistic Symbol Extension

- **To map $\mathrm{T}^2 \in C_2^2$ onto the original set of symbols x_i to transmit E^R in a unique way:**

$$\boldsymbol{e_i = t_i * x_i}$$

- **For PSK signals, this is just a phase shift of the original symbols.**

Probabilistic Symbol Extension

- **Decoding this mapping is trivial since Bob knows t_i through I.**

$$y_i = \sum_{i=1}^{2} r_i / t_i$$

- **The term $\frac{r_i}{t_i}$ de-rotates each symbol and adds them up to a symbol in the original constellation.**

Analysis

Obfuscation Efficacy

- **With this paradigm, it is possible to obfuscate a frame to have an arbitrary amount of bits**

$$L_R \approx \frac{L_D}{M}\left(1 + M\sum_{i=2} iP_r^i\right)$$

- **It is not yet clear how much obfuscation is necessary to prevent traffic analysis though.**

Encryption Multiplier

- Even when P_r^i is known, a brute-force attack on any encrypted of length L_D would need to make additional guesses.

- In the case of having exactly half the symbols be extended, this is around 10^{238} for a BPSK packet of 100 bytes.

Results

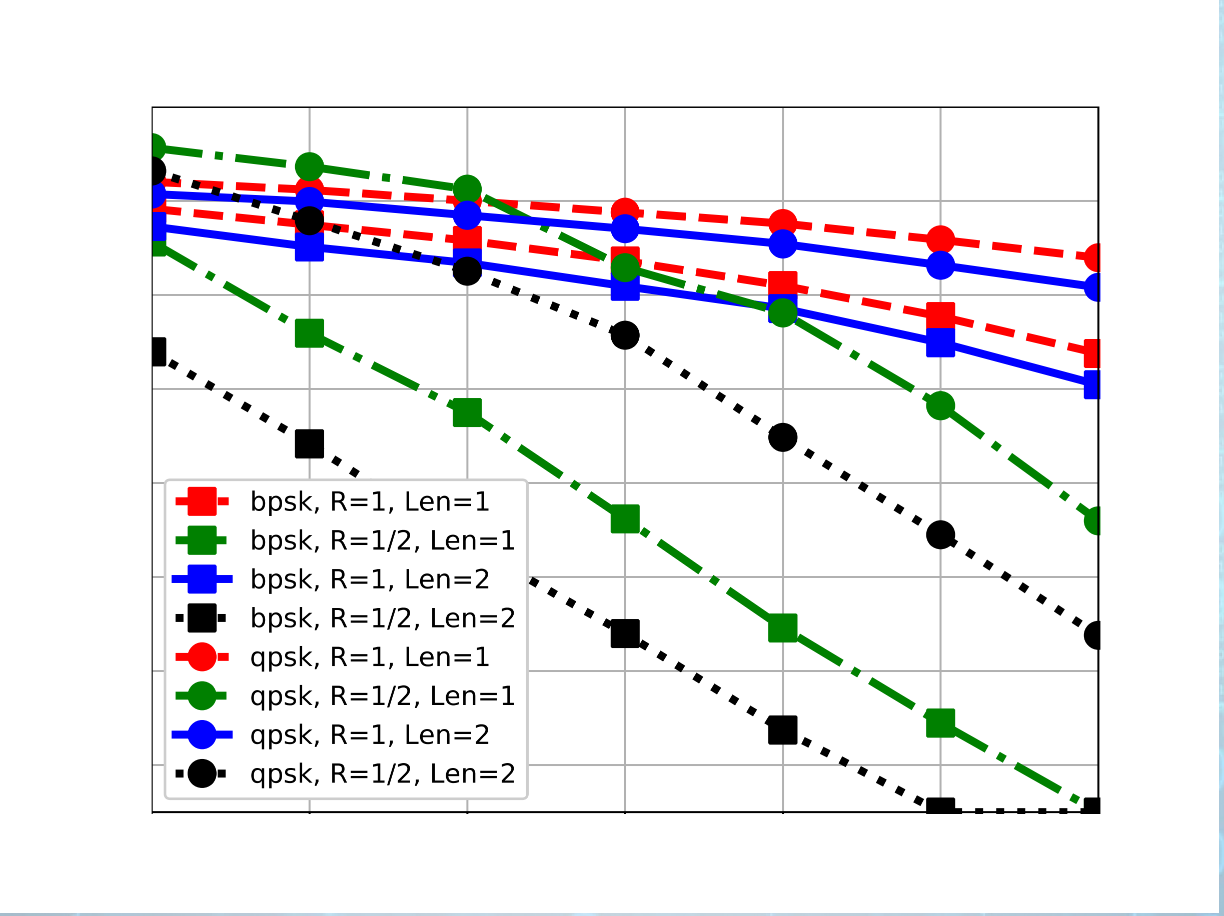
THE UNIVERSITY OF ARIZONA

PSE Data Rate

- Ideally, a signal could be obfuscated without a substantial loss in effective data rate
- PSE can partially make up for this as an extended symbol receives a benefit in SNR of:

$$\gamma_{extended} = 20 * log_{10}(R)$$

PSE Data Rate



Conclusion

THE UNIVERSITY
OF ARIZONA

Conclusion

- **The frame size, frame interarrival times, and frame direction are enough to compromise important security requirements**

- **PSE can be used to arbitrarily obfuscate frame size at the cost of effective data rate.**