I. Motivation

Jamming attacks have been studied as wireless security threats disrupting reliable RF communication in a wireless network. Existing countermeasures often make use of spread-spectrum techniques such as Frequency Hopping Spread Spectrum and Direct Sequence Spread Spectrum [3]. Communication parties rely on a pre-shared common key (hopping sequence or spread code) which is unknown to the jammer, thus making the system robust against jamming attacks. This, however, raises another issue: namely establishing the initial secure key pairing under jamming attacks. This demands another secure communication channel and this introduces circular dependency.

To break the dependency between anti-jamming communications and key establishment, recent proposals [4, 5] adopted a Pseudo-random Frequency Hopping (PFH) technique similar to that in Bluetooth. During the key establishment, a sender and a receiver randomly switch over multiple frequencies. Upon meeting on the same frequency by chance, they exchange a common key. To increase the probability of encounters, the sender hops faster than the receiver. In Bluetooth, a master randomly hops every 625μs, whereas a slave switches frequencies every 11.25ms; thus the hopping ratio becomes 18 (=\frac{625\mu s}{11.25ms}) . PFH, however, introduces considerable latency: it could take up to 100 seconds in the worst case [5]. Moreover, relying on random encounters among nodes cannot provide an upper bound of the time and cost to complete the key establishment phase.

To resolve the problem, we propose a Frequency Quorum Rendezvous (FQR) scheme for fast and resilient key establishment. FQR exploits a quorum system which allows each node to build a hopping sequence independently during the key establishment phase. The intersection property of the quorum system increases probability that a pair of nodes meet within a bounded amount of time, during which they share a common key for future spread spectrum communications. We validate the proposed scheme via extensive simulations and present its robustness and efficiency.

II. Frequency Quorum Rendezvous

To discuss a frequency hopping as a key establishment mechanism, we assume that time is divided into frequency hopping periods, each of which consists of \( t \) time slots. And suppose that there are \( N \) frequencies available in a wireless network.

A frequency hopping system is constructed by assigning frequencies to \( t \) time slots in one period and determining a frequency hopping sequence, \( X \):

\[
X = \{x_0, \ldots, x_t\} = \{(0, c_0), \ldots, (t-1, c_{t-1})\},
\]

where \( x_i \in X \) contains a tuple of (time slot index, frequency index) and \( c_i \in \{0, \ldots, N-1\} \) represents the frequency index at time slot \( i \) in a time period. Given two hopping sequences \( X \) and \( Y \), they are said to rendezvous if they have at least one element in common; \( x_i = y_i \) \((0 \leq i \leq t-1)\). If a pair of nodes selects the rendezvous sequences of \( X \) and \( Y \) respectively, then they are guaranteed to be on the same frequency at the same time at least once within a period.

II.A. Quorum System

A quorum system is a collection of subsets (quorums) of a given universal set, in which any pair of two subsets have at least one element in common [1]. The proposed FQR scheme exploits a cyclic quorum system [2] to construct a set of hopping sequences. This subsection gives a brief description on it.

**Definition 1.** Given a finite universal set \( U = \mathbb{Z}_N = \{0, 1, \ldots, N-1\}\) of \( N \) elements, a subset \( D = \)
Algorithm 1 FQR System Construction Algorithm

Require: \( N, \kappa, U = \mathbb{Z}_N \), and a quorum system \( Q \)

Ensure: Sending/receiving sequence \( X \) and \( Y \)

1: Select \( i \) randomly, where \( i \in U = \{0, \ldots, N - 1\} \)
2: Obtain a quorum \( G_i = \{g_0, \ldots, g_{\kappa-1}\} \), where \( G_i \in Q = \{G_0, \ldots, G_{N-1}\} \)
3: \( X = \emptyset \) and \( Y = \emptyset \)
4: for \( j = 0 \) to \( \kappa^2 - 1 \) do
5: \( m \leftarrow j \mod \kappa \)
6: \( n \leftarrow (j - (j \mod \kappa))/\kappa \)
7: \( x_j = (j, g_m) \), where \( g_m \in G_i \)
8: \( y_j = (j, g_n) \), where \( g_n \in G_i \)
9: \( X \leftarrow X \cup x_j \)
10: \( Y \leftarrow Y \cup y_j \)
11: end for
12: return \( X = \{x_0, \ldots, x_{\kappa^2-1}\} \) and \( Y = \{y_0, \ldots, y_{\kappa^2-1}\} \)

\( \{a_1, \ldots, a_\kappa\} \subset \mathbb{Z}_N, a_i \in \{0, \ldots, N-1\} \) and \( \kappa \leq N \)
is called a cyclic \((N, \kappa)\) difference set if for every \( d \neq 0 \pmod{N} \) there exist at least one pair of elements \((a_i, a_j)\) such that \( a_i - a_j \equiv d \pmod{N} \).

Definition 2. Given a \((N, \kappa)\) difference set \( D = \{a_1, \ldots, a_\kappa\} \subset \mathbb{Z}_N \), a cyclic quorum system constructed by \( D \) is \( Q = \{G_0, \ldots, G_{N-1}\} \), where \( G_i = \{a_1 + i, a_2 + i, \ldots, a_\kappa + i\} \pmod{N} \) and \( i = 0, \ldots, N - 1 \).

II.B. FQR System

Algorithm 1 constructs the FQR system by assigning frequencies computed by the quorum system to time slots. The following procedure presents the algorithm with an example by setting \( N = 7 \) and \( \kappa = 3 \).

1. Construct a universal set \( U = \mathbb{Z}_7 = \{0, \ldots, 6\} \) and determine a \((7, 3)\) difference set \( D, (\sqrt{7} \leq 3 \leq 7) \); (2) Construct a cyclic quorum system \( Q = \{G_0, \ldots, G_6\} \) from \( D \); (3) A node \( A \) selects a random number, e.g., 5 from \( U \) and obtains a quorum \( G_5 = \{5, 6, 1\} \) from \( Q \); (4) The following equation assigns a channel to the time slot \( j \) using the quorum \( G_5 = \{g_0, g_1, g_2\} = \{5, 6, 1\} \): \( x_j = (j, g_m) \) and \( y_j = (j, g_n) \), where \( m = j \mod \kappa \) and \( n = j - (j \mod \kappa)/\kappa \); (5) Repeat step (4) for all \( 9 = \kappa^2 \) time slots. This constructs a sending sequence \( X = \{(0, 5), (1, 6), (2, 1), (3, 5), (4, 6), (5, 1), (6, 5), (7, 6), (8, 1)\} \) and a receiving sequence \( Y = \{(0, 5), (1, 5), (2, 5), (3, 6), (4, 6), (5, 6), (6, 1), (7, 1), (8, 1)\} \); (6) A node \( B \) repeats step (4-5) with a selected quorum, e.g., \( G_3 = \{3, 4, 6\} \), and then, construct two hopping sequences \( X' \) and \( Y' \).

Figure 1: FQR with \((7, 3)\) difference set under \( \mathbb{Z}_7 \). They rendezvous on channel 6 at time slot 7.

III. Evaluation

The FQR system is implemented based on the cyclic quorum system using minimal \((N, \kappa)\) difference sets [2]. For comparison, we implement PFH with the hopping ratio = 20. A random hopping (RH) is also implemented, which is PFH with the hopping ratio = 1. A jammer randomly chooses a target frequency on which it jams for each time slot. Then, it jumps to another random frequency to attack without pause.

During experiments, we measure two metrics. Time-To-Rendezvous (TTR) is an average time for a pair of nodes to rendezvous. Rendezvous Frequency Concentration (RFC) indicates the degree of the rendezvous scheme of focusing on a subset of the available frequencies. If frequency hopping tends to rendezvous on specific frequencies, the jammer can abuse this knowledge to launch efficient attacks.

III.A. Experiments and Results

At the first experiment, we vary the number of potential frequencies \((N)\) from 10 to 100. In Figure 2(a), FQR shows a lower TTR than those of RH and PFH, which indicates that a sender meets a receiver quickly. The gaps become clear as \( N \) increases because nodes in RH or PFH randomly select frequencies. Surprisingly, RH and PFH produce similar TTR curves. This implies that increasing the hopping ratio has little impact on TTR performance.

A low RFC value in Figure 2(b) indicates that more frequencies are used for rendezvous; a jammer can hardly predict the hopping sequences. PFH presents the worst performance due to high hopping ratio: a receiver sits on one frequency for 20 time slots regardless of \( N \) values. Thus, for \( N < 20 \), rendezvous occurs only on the frequency selected by the receiver, which hugely degrades the RFC performance. FQR works better than RH as it takes \( \kappa \) adaptively along with increasing \( N \).
The second experiment increases the probability of jamming attack ($P$) from .1 to .9 in a wireless network having 11 potential frequencies. For instance, 90% of $P$ implies that one packet (or symbol) is successfully delivered with probability of 10%.

Both RH and PFH present TTRs increasing steeply as $P$ increases in Figure 3(a). On the contrary, FQR shows lower tail. Compared with RH and PFH, FQR improves the performance by around 40% on average and displays robustness against various severity. Figure 3(b) represents RFC performance with varying $P$. FQR shows the best performance. All the curves gradually decline as $P$ increases, and with high probability of attacks (i.e., 90%), they show similar performance.

IV. Conclusion

We have presented a novel Frequency Quorum Rendezvous (FQR) scheme that achieves fast and resilient key establishment, making wireless communication more robust against jamming attacks. The nodes to hop over random multiple frequencies without a prior knowledge of communication partners. Using a quorum system increases the possibility that the nodes would meet within a bounded time. The experimental results showed that the proposed scheme outperforms existing methods in terms of TTR and RFC.

To support realistic scenarios, we consider mobility and multi-hop routing. For instance, if a pair of nodes moves, the TTR must be much smaller than the pair “contact” time, i.e., link longevity. Developing a multi-hop routing protocol in mobile environment will be another integral part of our future work.

References


