Тема

Подмена маяка для девайсов класса B в сетях LoRaWAN

Автор

Мазяр Алексей Николаевич, 21213

Введение

Технологии интернета вещей были внедрены для многих целей, в частности для передачи данных в автономных датчиках экологического наблюдения и коммунальном хозяйстве. Проблема состоит в том, что автономные датчики используют батарейное питание, из-за чего энергозатратная пересылка данных (особенно на большие расстояния) быстро выводит устройства из строя.
Одно из решений - протокол физического уровня, LoRa (long range). Основные его свойства - низкое энергопотребление и большая дальность передачи сигнала (до 15 км). В то время LoRa отвечает за низшие уровни, такие технологии как LoRaWAN отвечают за верхние (канальный и сетевой в случае LoRaWAN).

Краткое описание классов устройств LoRa

Каждое устройство, поддерживающее LoRa, принадлежит к одному из трёх классов: A, B, C. Все устройства выполняют операции класса A, устройства класса B выполняют операции классов A и B, устройства класса C выполняют операции всех классов.
Отличие устройств класса A от устройств класса B состоит в том, что первые ожидают сообщений по downlink только после заданного времени, отсчитанного от последнего сообщения по uplink. Такое поведение нужно, чтобы устройство могло спать, экономя энергию, не проверяя downlink лишний раз. Это вызывает проблемы, если сервер захочет послать сообщение устройству, не дожидаясь его uplink; ведь устройство просто будет спать и сообщение не будет сразу получено.
Для таких случаев используются устройства класса B. На эти устройства сервер периодически посылает фреймы-маяки. Эти фреймы содержат информацию о том, с какой периодичностью следует слушать downlink. Синхронизируя таким образом отправку и принятие сообщений, мы получаем достаточно энергоэффективную систему, так как если по пробуждении устройство не замечает начало фрейма-маяка, оно сразу засыпает.

Описание проблемы обеспечения безопасности

Спецификация LoRa диктует, чтобы фреймы-маяки рассылались broadcast’ом, а устройства воспринимали их из любого источника. Поэтому нельзя определить пришёл ли фрейм из надёжного источника, и хакер может отправлять собственные ложные фреймы. С их помощью можно сдвинуть период окна, когда устройство слушает downlink. Тогда можно будет минимизировать пересечение окон ожидания у устройства и отправки у сервера. Пока хакер продолжает транслировать поддельные фреймы-маяки, устройство не сможет получать сообщения по downlink.

Атака с использованием уязвимости

Периодическая отправка ложных фреймов - не сложная задача, а переконфигурация устройства на новый интервал - нетривиальная проблема.

1. Сначала хакер узнаёт, когда рассылаются настоящие фрейм-маяки.
2. После этого хакер будет отсылать свои фреймы, сдвигая фазу на некоторый интервал. Если интервал сдвига достаточно мал, устройство будет сдвигать своё окно ожидания сообщений по downlink вместе с фреймами хакера.
3. Когда суммарная продолжительность этих сдвигов станет равной продолжительности прослушивания downlink, сдвиги прекращаются.
4. Поддельные фреймы приходят естественно не одновременно с настоящими, из-за чего устройство может избрать настоящие фреймы, как конфигурацию. Чтобы этого избежать, следует добавить мусорный payload, который будет отправляться вместе с поддельным фреймом. Если правильно настроить сдвиг поддельных фреймов, этот payload будет приниматься устройством тогда же, когда приходят настоящие фреймы-маяки. Благодаря этому шанс, что устройство их воспримет, уменьшается.

Защита от атаки

Использование авторизации маяков

Надёжным способом избежать вмешательства в отправку фреймов-маяков является аутентификации сообщения, например с использованием имитовставки. Тогда фреймы хакера просто отбрасывались бы и атаку нельзя было бы начать. Это решение плохо тем, что спецификация требует, чтобы устройства воспринимали все фреймы.

Использование счётчика фреймов

Все LoRa устройства ведут счётчики, отправленных по uplink и полученных по downlink фреймов. Значения этих счетчиков прикладывается к каждому фрейму, а при получении очередного фрейма устройство или сервер сравнивает значения полученных счётчиков с предыдущими: если полученные значения меньше, то фрейм откидывается. Конечно, хакер может узнать значение счётчиков, проверяя настоящий фрейм-маяк. Если устройство получит фрейм от сервера, оно заметит отставание по счётчику и сможет предупредить об этом сервер. Такие предупреждения могут стать основанием полагать, что устройство было атаковано. В таком случае сервер может заставить устройство перейти в режим работы класса A или C, что полностью защитит его от подобной атаки.
Режим работы класса C слушает downlink почти всегда. Прослушивание останавливается только при отправке сообщений по uplink. Этот режим менее энергоэффективен других.

Следует отметить, что на текущий момент эта уязвимость не защищена, однако атаки с их использованием не были зафиксированы в полевых условиях. И всё же были удачные опыты по воспроизведению таких атак в искусственных сетях.