Безопасность одноранговых сетей

Peer-to-Peer (P2P) Network Security

Минко В.С. vitaly.minko@infotecs.ru
Отдел ПиР ПАК

28.12.2014

«Недостаточно иметь систему, где каждому (включая саму систему) нужно доверять. Система также должна быть устойчива против внутреннего нарушителя!» – Роберт Моррис, бывший научный руководитель АНБ.

Peer-to-Peer Systems and Security

В строгой (англ. pure - все узлы «равны») одноранговой системе, каждый участник — потенциальный нарушитель

- Ни одному другому пиру нельзя доверять ни в чём.
- Нет центров сертификации, и т. п. доверительных центров.
- Обеспечение любого рода безопасности очень сложно!

Typical Adversary Models

- Global Passive Adversary (GPA)
  - Слушает и анализирует всю сеть
  - Нет активного вмешательства в сеть
- Global Active Adversary (GAA)
  - Помимо GPA также производит активные атаки
- Partial Passive Adversary (PPA)
  - Слушает только часть (<<50%) сети
- PPA или GPA с несколькими активными узлами

Cryptographic Primitives

- Генератор случайных чисел
- Хеширование
- Симметричное шифрование
- Асимметричное шифрование

Security Goals

- Доступность
- Секретность
- Целостность
- Подлинность

В некоторых случаях требуется обеспечить лишь некоторые из них. Например, для открытых сетей не требуется секретность.

P2P Authentication

- Как производить аутентификацию в строгой P2P системе (без глобального центра сертификации)?
- Обычно аутентификация — первый шаг в достижении стандартных целей безопасности.
- Многие другие цели безопасности бессмысленные без этого – т. е., нет секретности, если мы случайно раскроем секрет не тому получателю.

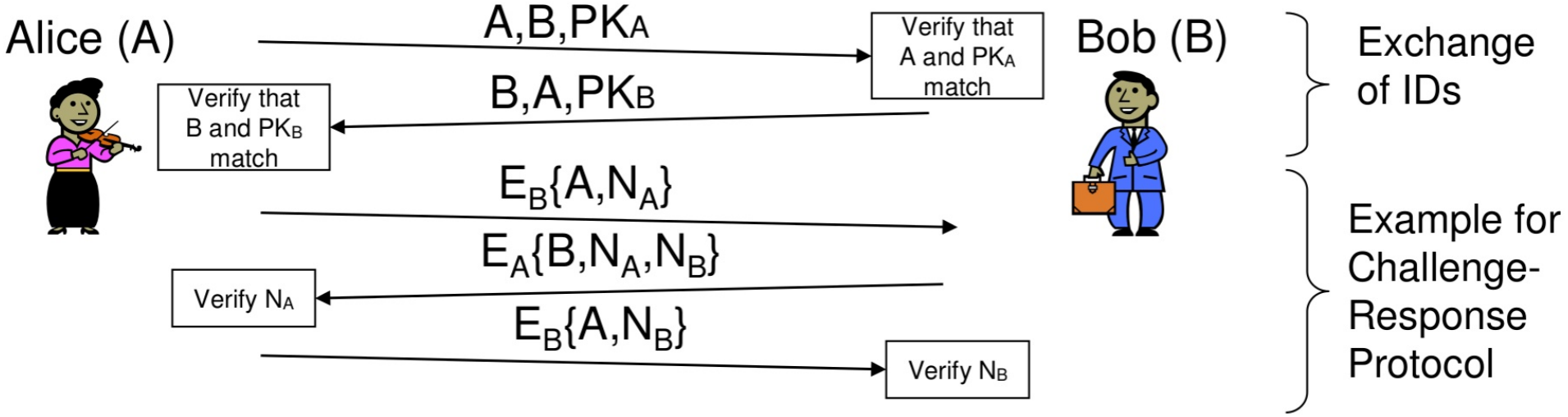
P2P Authentication

- Public key ≡ identity (IDx := H(PKx))
- Алиса может подписывать свои сообщения: A, PKa , Sa(M)

Такие идентификаторы называются криптографическими идентификаторами (“cryptographic identifiers” или self-certifying identifiers).

Cryptographic Identifiers

Узлы могут проверить ID друг друга, используя простой протокол вызов-ответ (challenge-response protocol):



Устанавливает, что A и B взаимодействуют с владельцами верных ID и нет злоумышленника по середине (man-in-the-middle). Случайные последовательности (Nonce) Na и Nb используются как вызов.

Boyd’s Theorem

Теорема №1: „Предположим, что у пользователя есть либо конфиденциальный канал к нему, либо аутентифицированный канал от него при некотором состоянии системы. Тогда в предыдущем состоянии, такой канал также должен был существовать. Индуктивным выводом получается, что такой канал существовал во всех предыдущих состояниях.“

„Другая интерпретация теоремы — никакой безопасный канал не может был сформирован между пользователями, которые уже не обладают общим секретным ключом*. Вывод вполне естественный — нельзя получить что-то из ничего.*“

Boyd’s Theorem

Теорема №2: „Безопасное соединение между двумя пользователями может быть установлено путём последовательности безопасных передач ключей, если существует доверенные цепочки от одного пользователя к другому и обратно.“

Можно ли получить безопасную внутрисистемную аутентификацию без CA или TTP и без предварительных контактов между пользователями?

- Нет.
- Единственный путь обойти CA или TTP — это взаимодействие по другим канала (out-of-band communication).

Zfone Authentication (ZRTP)

Идея: комбинировать подтверждение через человеческое взаимодействие и подход «baby duck»

- A и B производят обмен ключами по Diffie-Hellman (обмен числами g^a и g^b и создание разделяемого секрета g^ab).
- Ключевой материал прошлых сессий используется согласно подходу «baby duck» (как в SSH).
- Создаётся *Short Authentication String* (SAS) как хеш от чисел Diffie-Hellman-а.
- Оба пользователя зачитывают SAS и голос передаётся другому пользователю. Если прочитанный номер верен, пользователь подтверждает аутентификацию.

Злоумышленник по середине должен перехватить и изменить числа Diffie-Hellman лишь при первом взаимодействии. Поэтому, невозможно провести стандартную атаку MiTM.

Trust vs. Authentication

В *открытых* P2P-сетях, нас больше интересует не кто оперирует другим узлом, нам интереснее знать каково будет его поведение:

- Будет ли пир следовать протоколу сети?
- Будет ли пир разделять ресурсы (таки как файлы в Bittorrent-сетях)?

Мы никогда не можем быть уверены, что пир будет нашим «другом».

Reputation

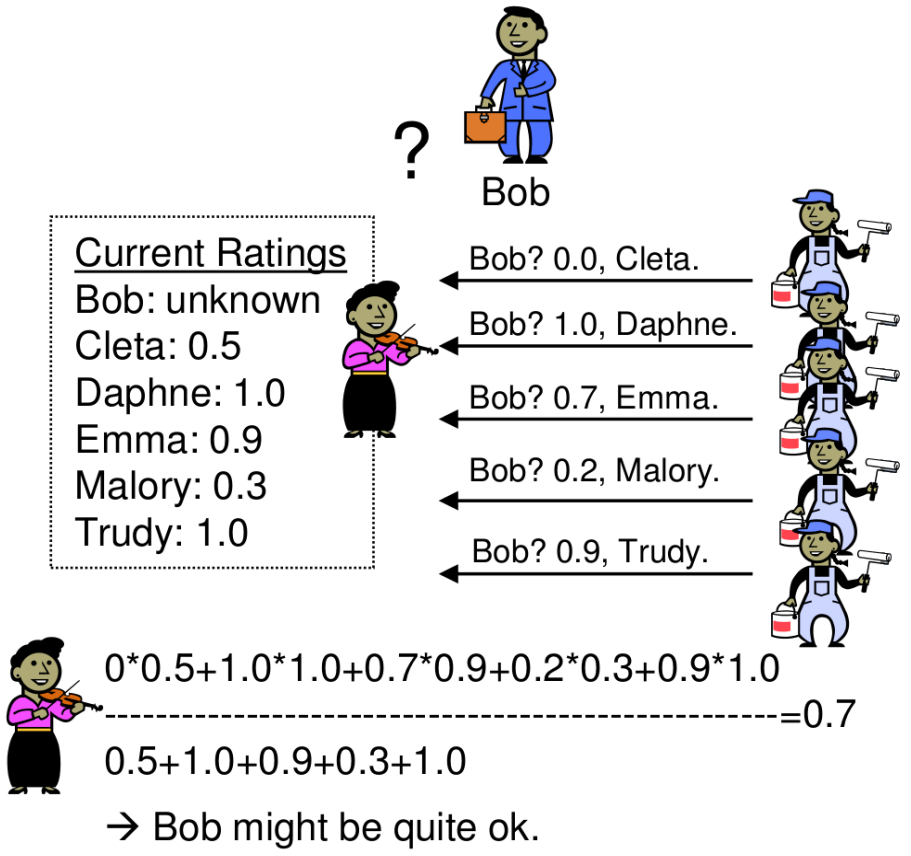
- Доверие (Trust) пиру (выраженное, как правило, численно), основанное на опыте или предварительных данных
- Глобальная репутация: одно принятое значение на каждого из пиров
- Локальная: каждый узел сам вычисляет значение для каждого другого узла

Для определения репутации требуется проводить:

- **Наблюдение** за действиями других узлов (мы успешно скачали желаемый ресурс от Алисы)
- **Оценку** действий других узлов (Алиса нам содействовала)
- **Хранение и сбор** результатов оценки (увеличить счётчик для Алисы на сервере репутаций)
- **Прогнозирование** будущих поведений узлов (Алиса содействовала в 57% случаев)

Decentralized Reputation

- Использовать свой личный опыт
- Использовать рейтинги других пользователей для вычисления комбинированного значения
- Для вычисления рейтинга неизвестных ранее узлов, использовать средневзвешенное значение доверия с доверием в качестве меры веса (web-of-trust - сеть доверия)



Attacks on Reputation

- **Непостоянство** — злоумышленник может действовать хорошо некотрое время, а потом изменить поведение
- **Отбеливание** — узлы с негативным рейтингом уходят и возвращаются с новыми «невинными» идентификаторами
- **Сговор** злоумышленников — узлоумышленники дают друг другу хорошие рейтинги

Sybil Attack

- Добавлять узлы в сеть несколько раз, каждый раз с новым идентификатором
- Потенциальные цели:
  - Помочь в проведении других атак, расположив узлы в заданных местах (например, атаки на порчу таблицы маршрутизации).
  - Атаки на репутацию
  - Атаковать связанность сети
  - Атаковать дублированные данные (например, в DHT)

Sybil Defense: Douceur

- Использовать аутентификацию с TTP, которая ограничит создание идентификаторов
- Использовать “внешние” идентификаторы (IP-адрес, MAC, e-mail)
- Использовать “дорогие” идентификаторы (требовать поиск proof-of-work, требовать плату за регистрацию)
- Прямая валидация идентификаторов (одновременно проверять все идентификаторы)

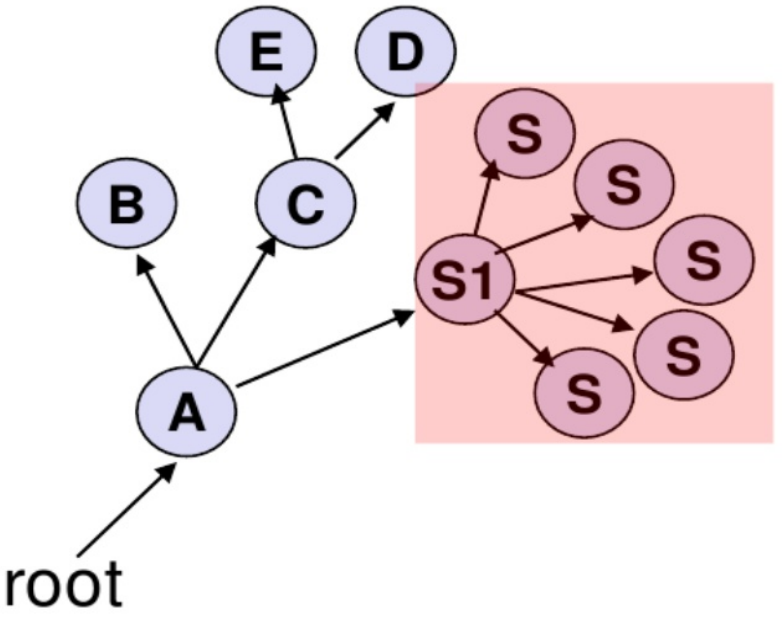
John Douceur:

- Использование ограничения по ресурсам для предотвращения Sybil-атаки требует невыполнимых (нереалистичных) условий
- Нет другого способа полностью предотвратить Sybil-атаку кроме централизованного сервиса удостоверения идентификаторов.

Sybil Defense: The Bootstrap Graph

Предположение:

- Первый Sybil-узел входит в сеть через произвольный входной узел (bootstrap node)
- Остальные Sybil-узлы входят в сеть через другой Sybil-узел

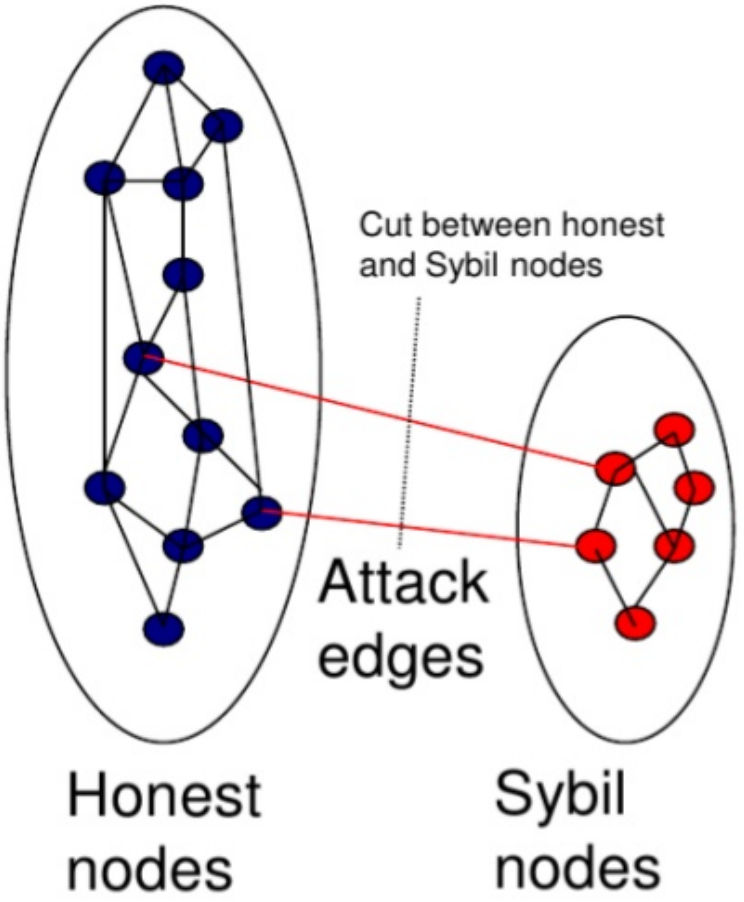


Sybil Defense: The Bootstrap Graph

- Идея: при выборе пиров, выбирать узлы из разных поддеревьев графа
- Входные ноды должны иметь политики доступа, например, основанные на социальных связях (Friend-to-Friend).

Sybil Defense: SybilGuard

- Узлы представляют персоналии из «реального мира», а ребра — социальные связи между узлами (т. е. отношение вида «знает», «доверяет», «друг», и.п.).
- Поскольку Sybil-узлы соответсвуют только нескольким персоналиям, у них будет меньше ребер с другими группами, чем у честных нод.
- Идея: Использовать пересечения случайных маршрутов для определения является ли нода в подграфе честных нод или в подграфе Sybil-нод.



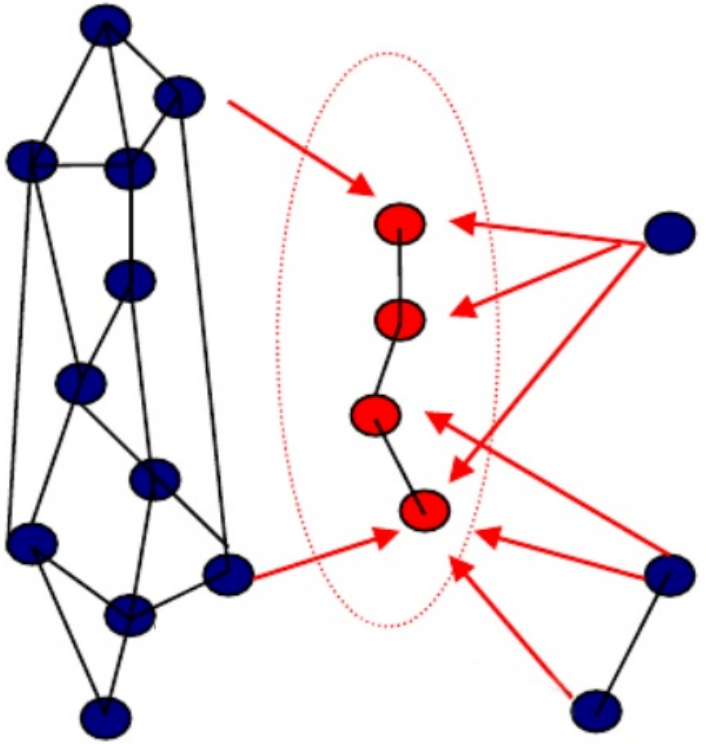
Poisoning Attacks

Злоумышленник использует ложную информацию, чтобы сломать целостность системы:

- **Index** Poisoning (перенаправлять ноды, запросившие определенный ресурс на ноды злоумышленника, связывать мета-данные с неверными данными (например, в DHT));
  - Предусматривать проверку подлинности данных
- **File** Poisoning (Захламление сети ложными или порченными файлами);
  - Использовать Web-of-trust (сеть доверия)
- **Routing Table** Poisoning (Внедрение нод злоумышленника в таблицу маршрутизации, основываясь на структурных ограничениях сети).
  - Использовать альтернативные неоптимальные маршруты

Eclipse Attack: Goal

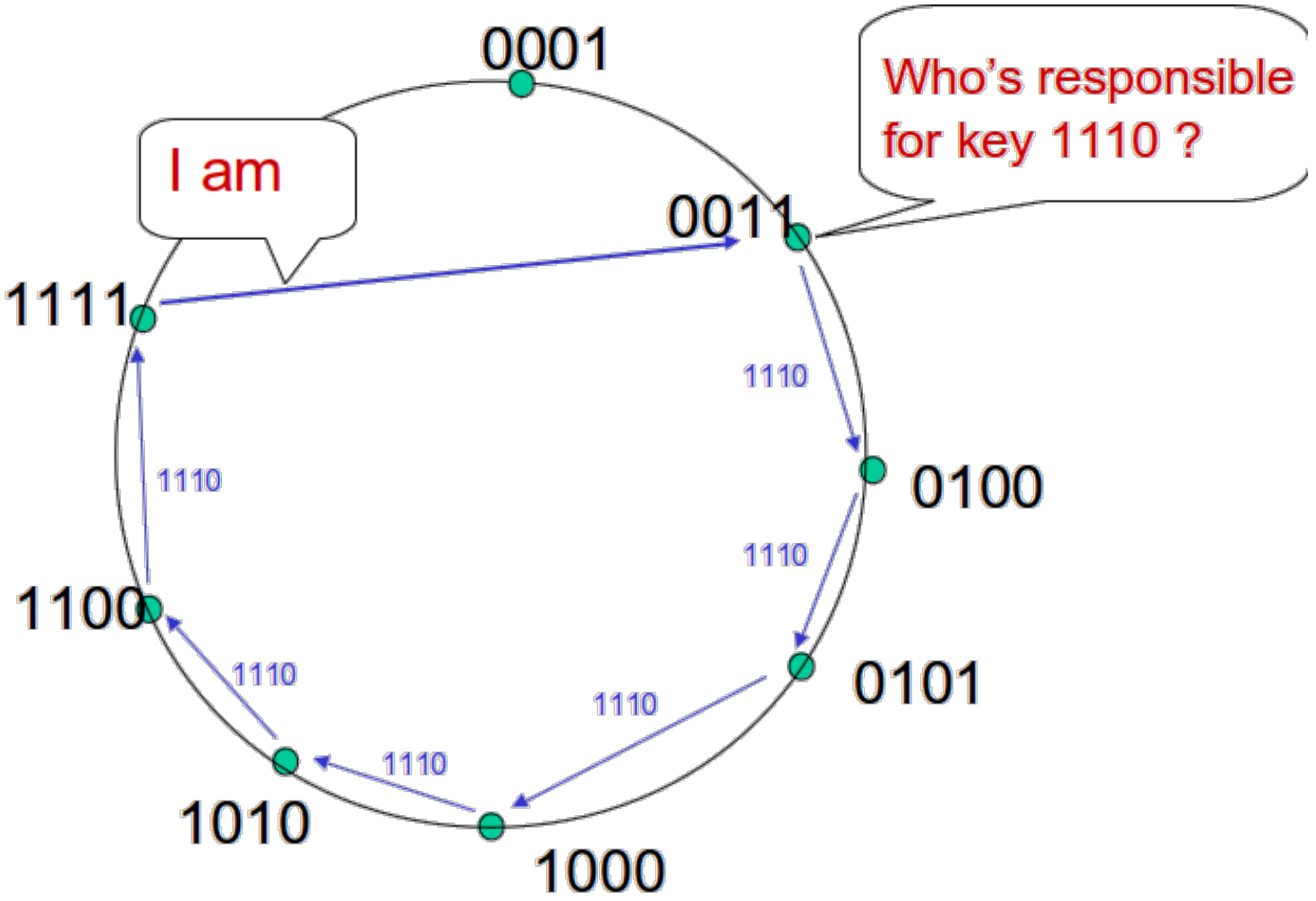
- Отделить ноду или группу нод от остальной сети
- Изолировать узлы (DoS, слежка) or изолировать данные (цензурирование)



Eclipse Attack: Techniques

- Использовать обнаружение соседних нод и ограничения таблицы маршрутизации для позиционирования нод злоумышленника (детали зависят на структуре оверлейной сети).
- Использовать Sybil-атаку для увеличения числа нод злоумышленника.

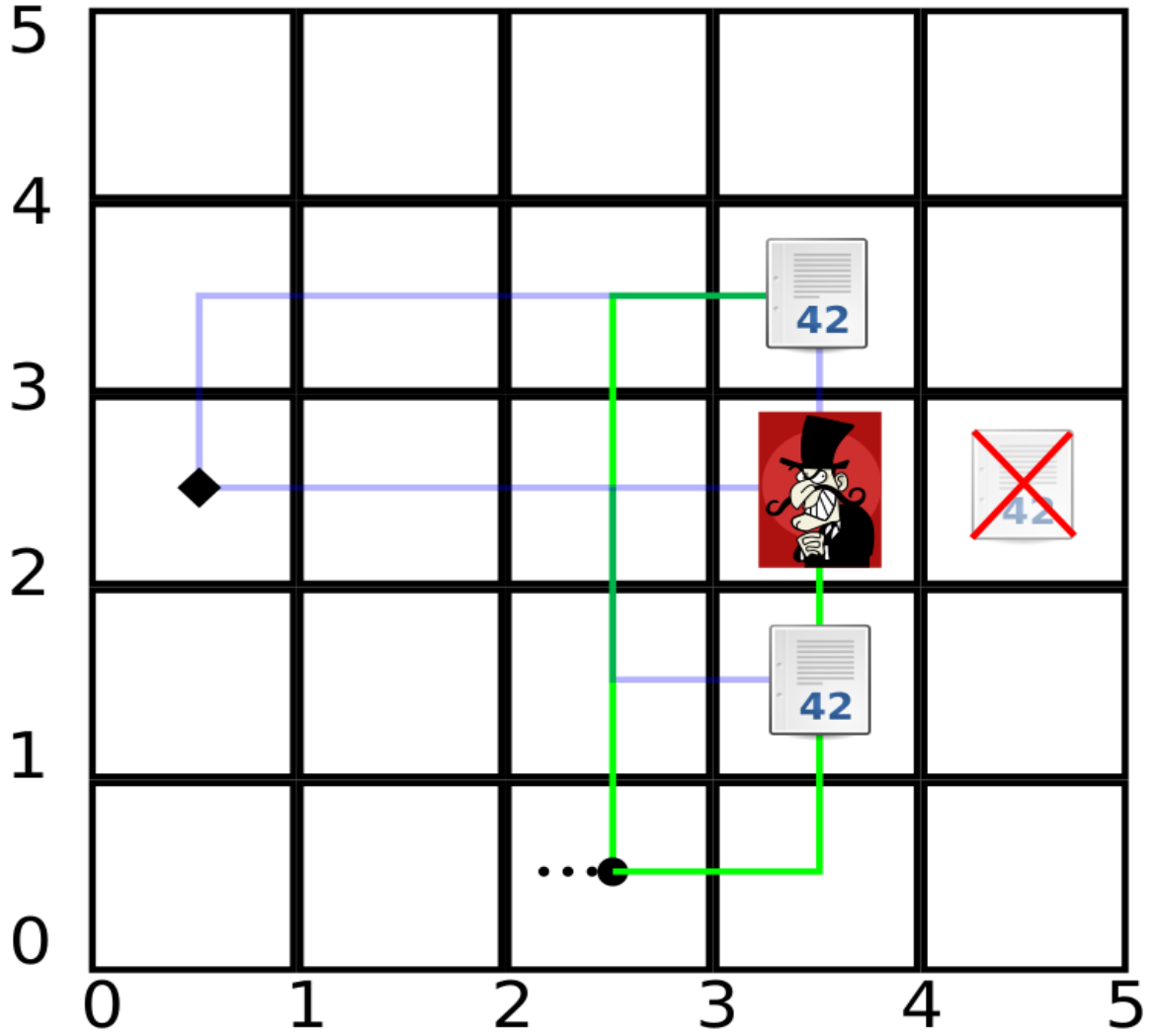
How DHT works



Eclipse Attack: Defenses

- Использовать большое число связей
- Репликация данных
- Разнообразие при выборе соседних узлов (из разных подсетей IP, разное географическое расположение)
- Постоянный поиск новых узлов (“continuous” bootstrap)
- Контроль поведения соседних пиров (по возможности)
- Предпочитать долгоживущие связи / старые пиры

Eclipse Attack in R5N DHT

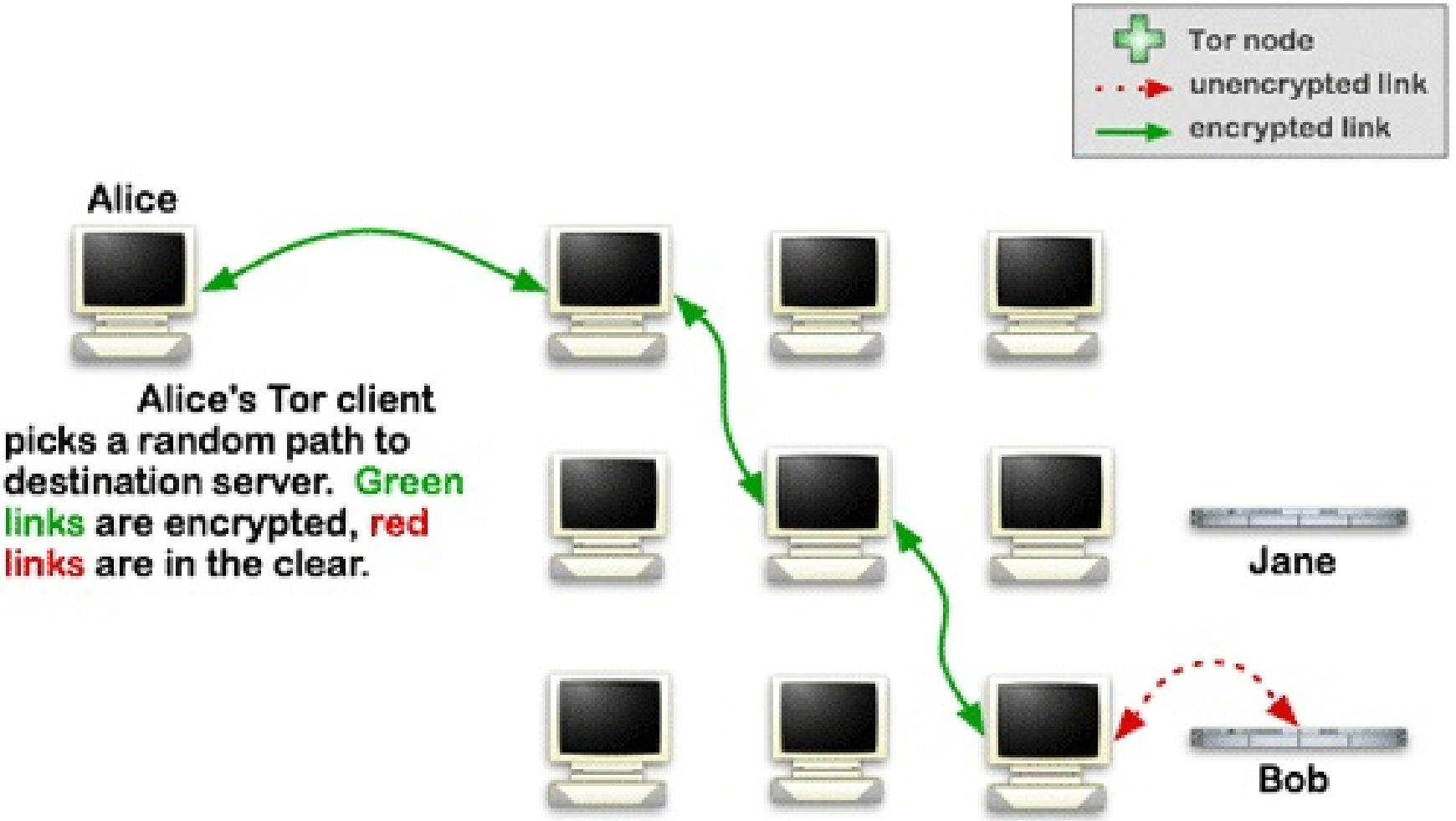


Timing Attacks

Класс атак на анонимные сети.

Рассмотрим на примере сети Tor.

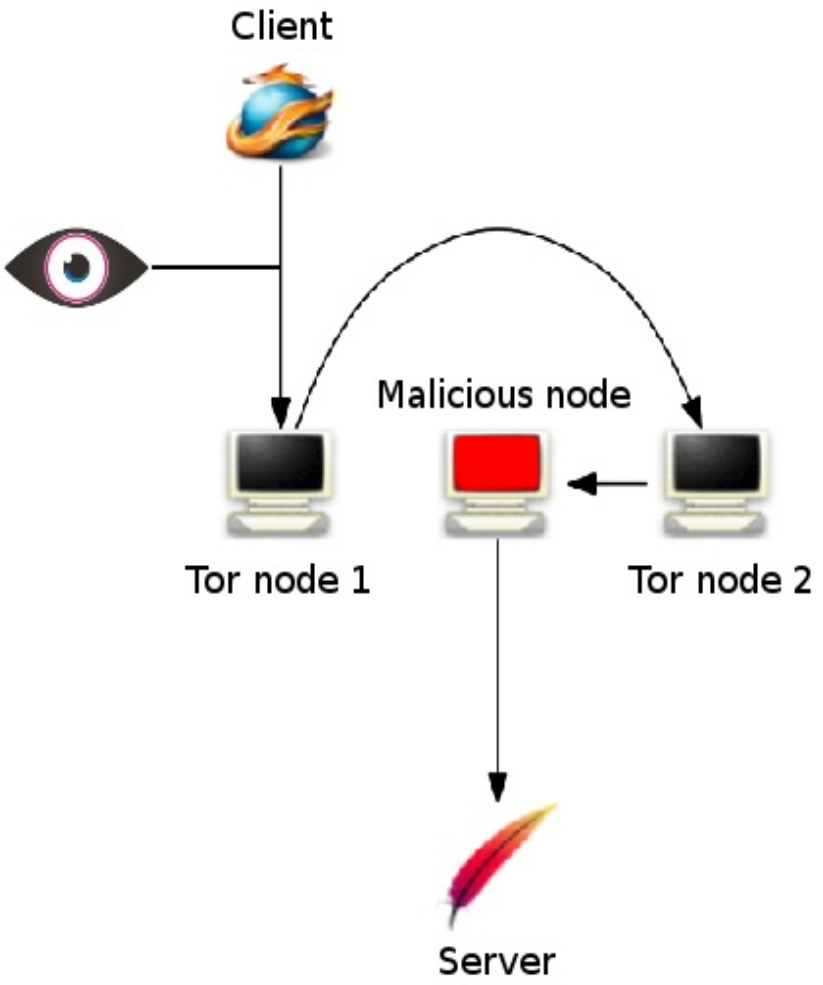
How Tor works



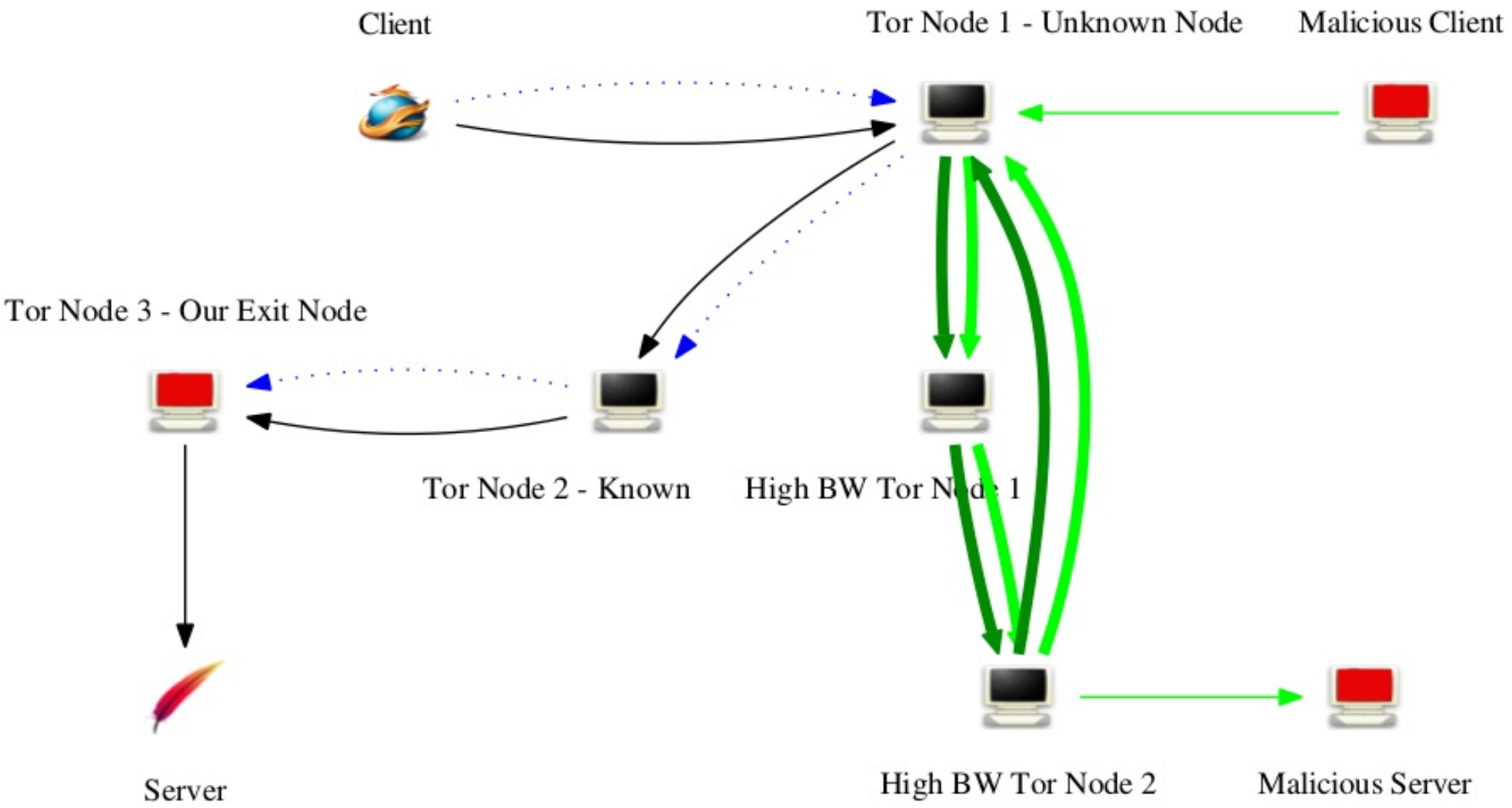
Timing Attacks: Idea

Идея: Пересылать сообщения, используя определённый шаблон по временным задержкам. Измерять латентности в передаче данных, коррелировать их для определения источника данных.

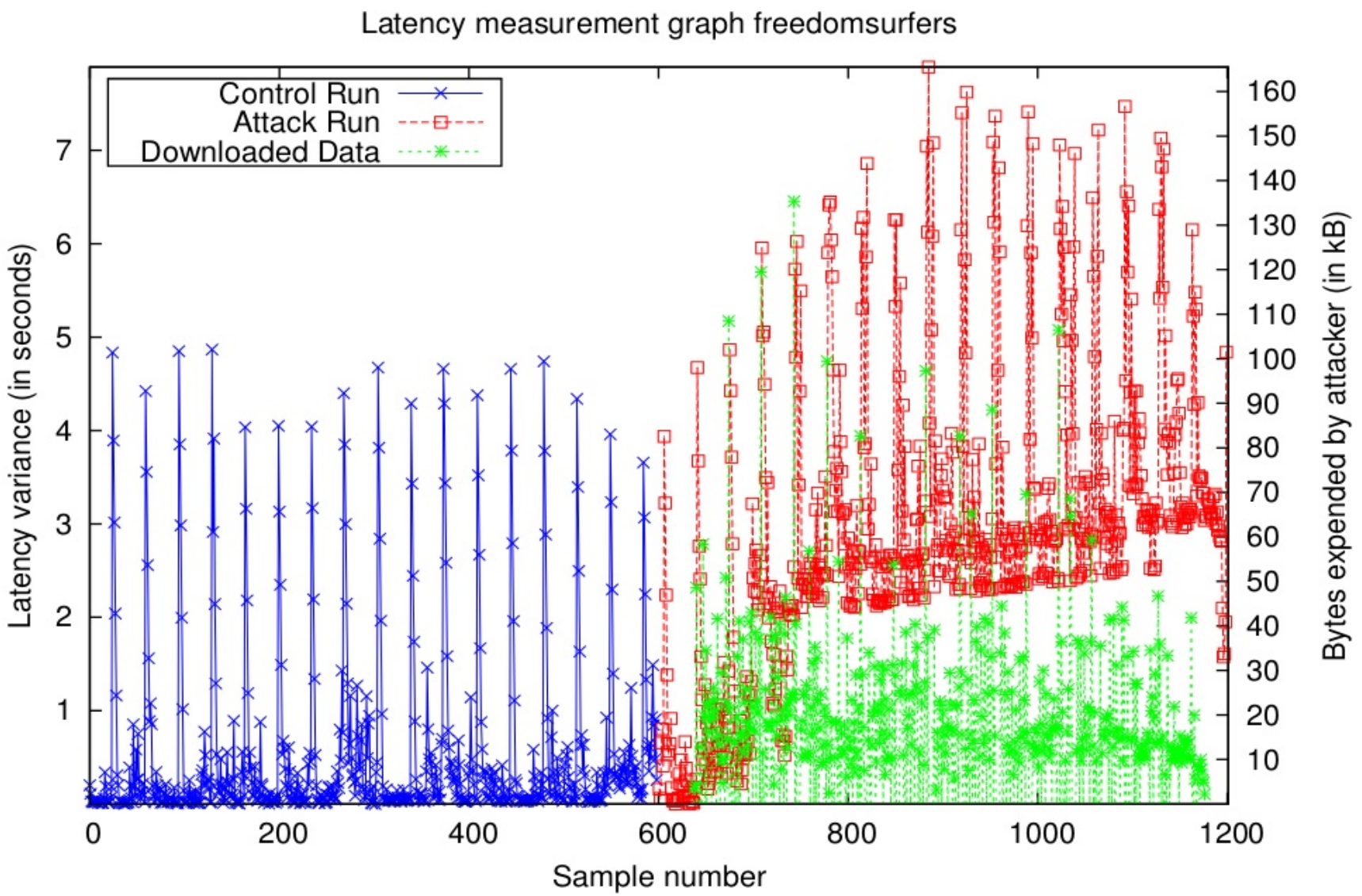
Timing Attacks: Tor example



Timing Attacks: Circular Path



Timing Attacks: Gathered Data Example



Timing Attack: Defenses

Противодействие:

- Вносить задержку в передачу сообщений
  - Не подходит для низколатентных анонимных сетей (Tor, I2P). Они обычно уязвимы к GPA, GAA

Questions?

?