Анонимная сеть «Hidden Lake»

Коваленко Геннадий Александрович

«Hidden Lake» (HL) — это децентрализованная анонимная F2F (Friend-to-Friend) сеть с теоретической доказуемостью на базе очередей (QB-задача)

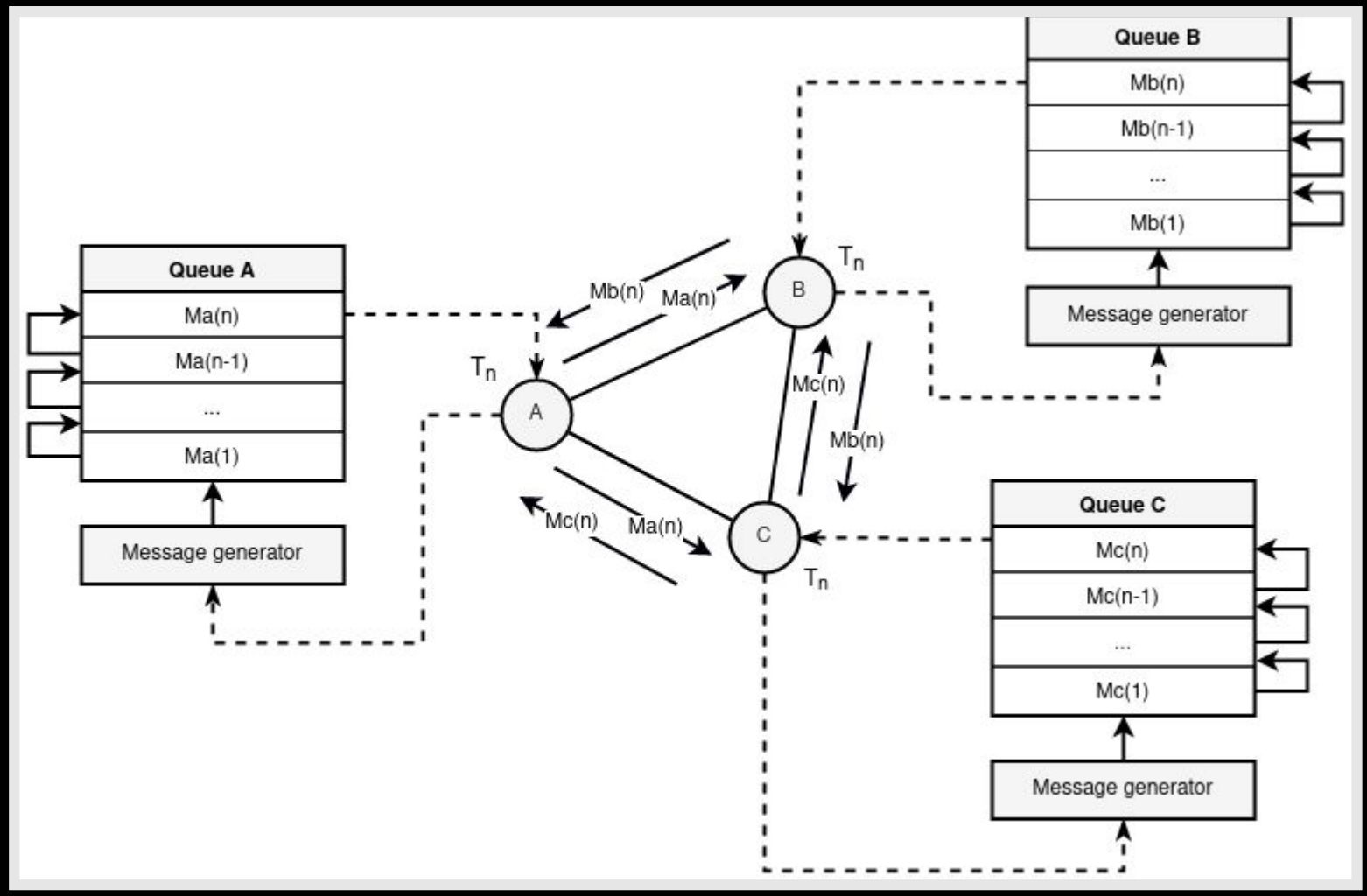
«Hidden Lake» является **Friend-to-Friend** сетью. Данное свойство определяет специфичный вид соединения участников в системе посредством ручной установки списка друзей

Задача на базе очередей (QB-задача)

простыми словами

1. Каждое сообщение шифруется ключом получателя,
2. Сообщение отправляется в период = **T** всем участникам сети,
3. Период **T** одного участника независим от периодов $\mathbf{T_1}$, $\mathbf{T_2}$, ..., $\mathbf{T_n}$ других участников,
4. Если на период **T** сообщения не существует, то в сеть отправляется ложное сообщение без получателя,
5. Каждый участник пытается расшифровать принятое им сообщение из сети.

Задача на базе очередей (QB-задача)



Задача на базе очередей (QB-задача)

формальным языком

Система:

$QB\text{-}net = \Sigma^{n}_{i=1}(T = \{t_i\}, K = \{k_i\}, C = \{(c \in \{E_{kj}(m), E_r(v)\}) \leftarrow^{ti} Qi\})$

Состояния:

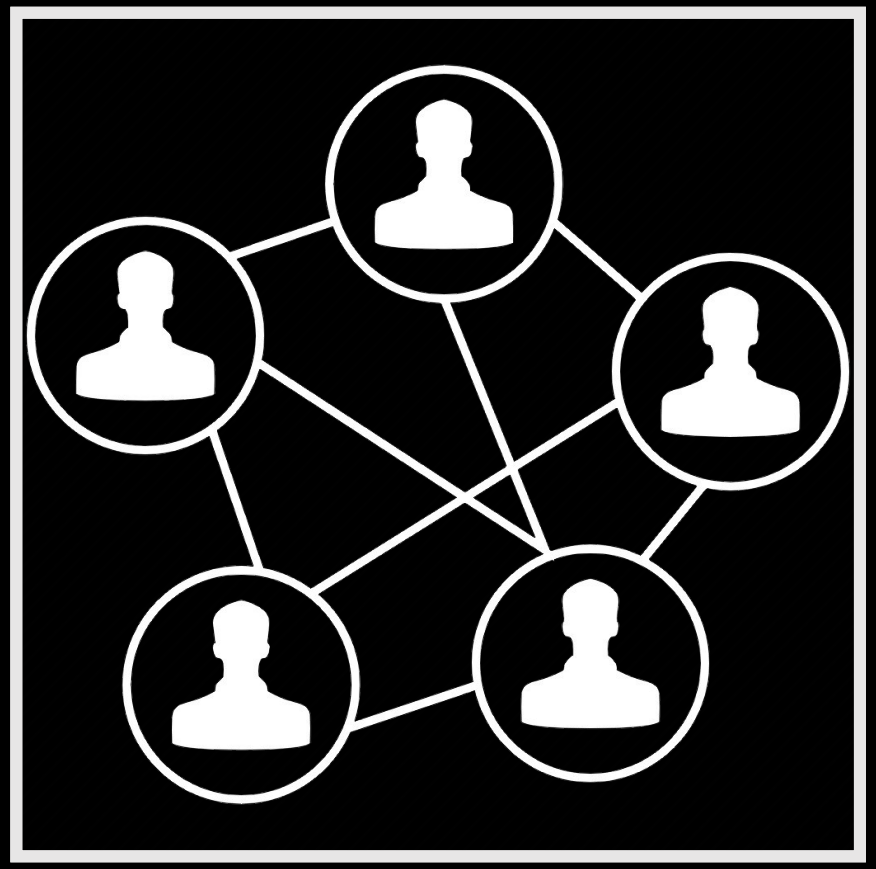
1. $Q \leftarrow (c = E_{ki}(m))$, где $k_i \in K$, $c \in C$,
2. $(c = E_{ki}(m)) \leftarrow^{t} Q$, если $Q \neq \emptyset$, где $t \in T$, $k_i \in K$, $c \in C$,
3. $(c = E_r(v)) \leftarrow^{t} Q$, если $Q = \emptyset$, где $t \in T$, $r \notin K$, $c \in C$,
4. $m' = D_k^{-1}(c)$, где $c \in C$

Сравнение с другими задачами анонимизации

	QB	EI	DC	Onion	Proxy
Теоретическая доказуемость	+	+	+	-	-
Накопительный эффект анонимности	-	+	-	-	-
Полиморфизм информации	-	+	+	+	-
Вероятностная маршрутизация	-	+	-	+/-	+/-
Периодичность генерации сообщений	+/-	-	+	-	-
Независимость анонимности от связей	+	-	-	-	-
Простота масштабирования	-	-	-	+	+
Простота программной реализации	+	-	-	+	+
Стадия анонимности	5^	6	1^	4 или 6	3
Сеть-представитель	Hidden Lake	-	Herbivore	Tor	Crowds

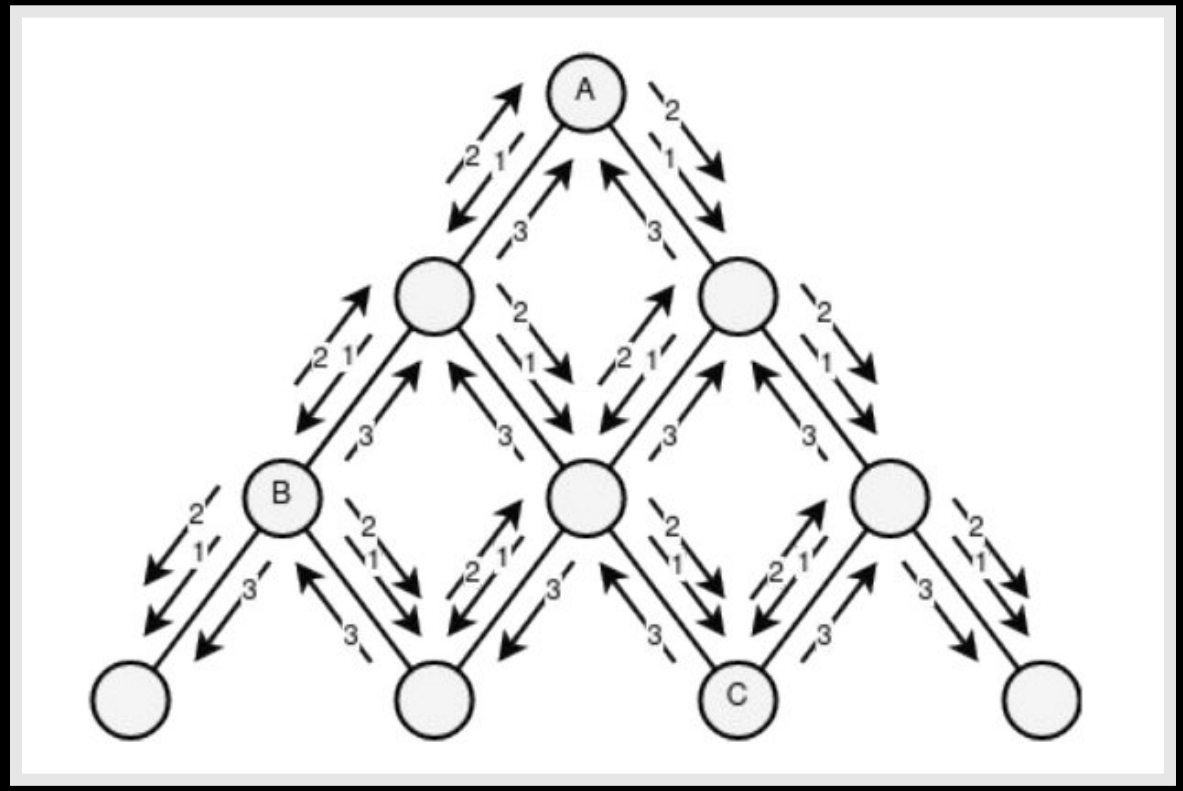
«Hidden Lake» относится к **абстрактным** анонимным сетям, которым не важны такие критерии, как:

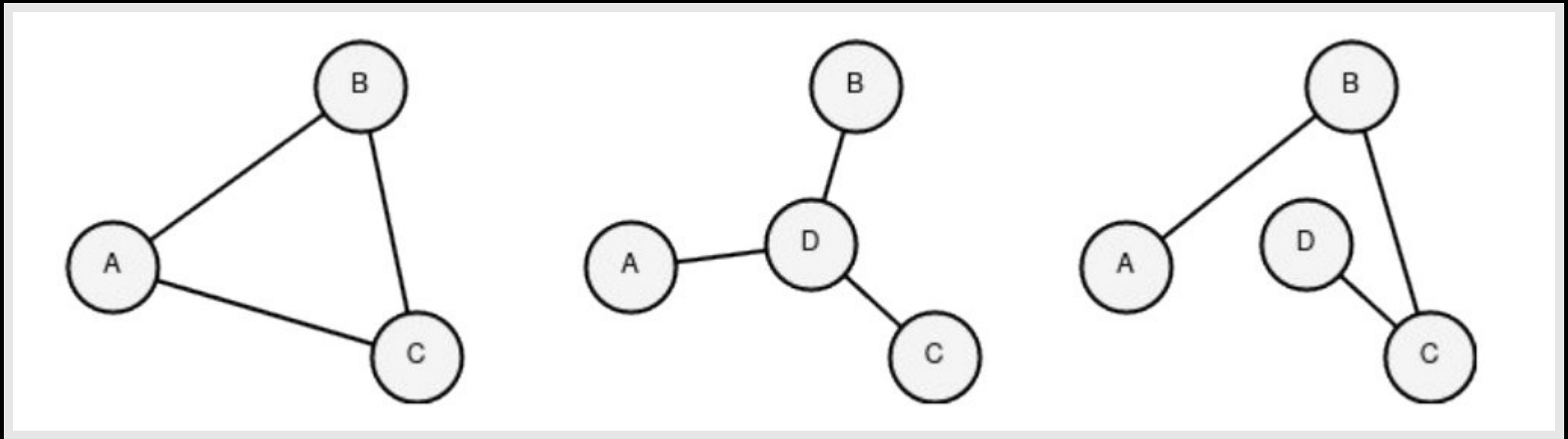
1. уровень централизации
2. количество узлов
3. расположение узлов
4. связь между узлами



За счёт своей абстрактности сеть «Hidden Lake» способна формировать **тайные каналы связи** с анонимизирующим свойством даже внутри централизованных сервисов

Главным недостатком сети является **линейная нагрузка** на систему, зависимая от количества участников



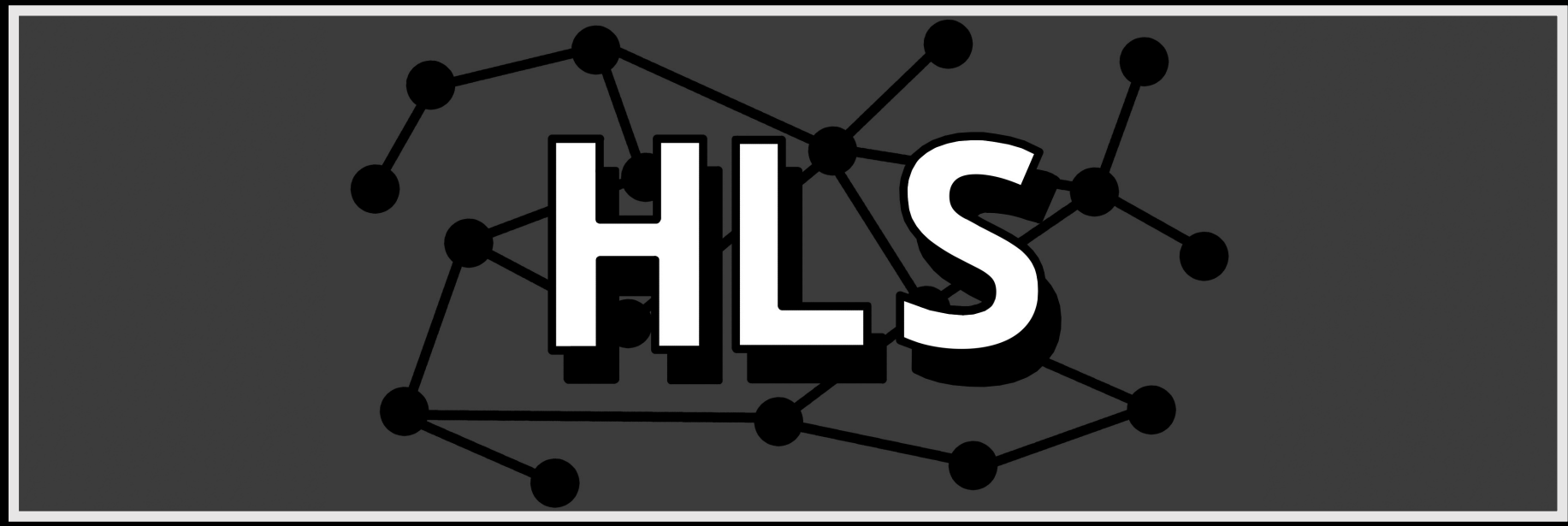


Частичным решением проблемы линейной нагрузки стало создание обособленных друг от друга **«малых озёр»** (сетей) посредством применения **сетевого ключа**

Философия разработки сети «Hidden Lake» основывается на **микросервисной** архитектуре

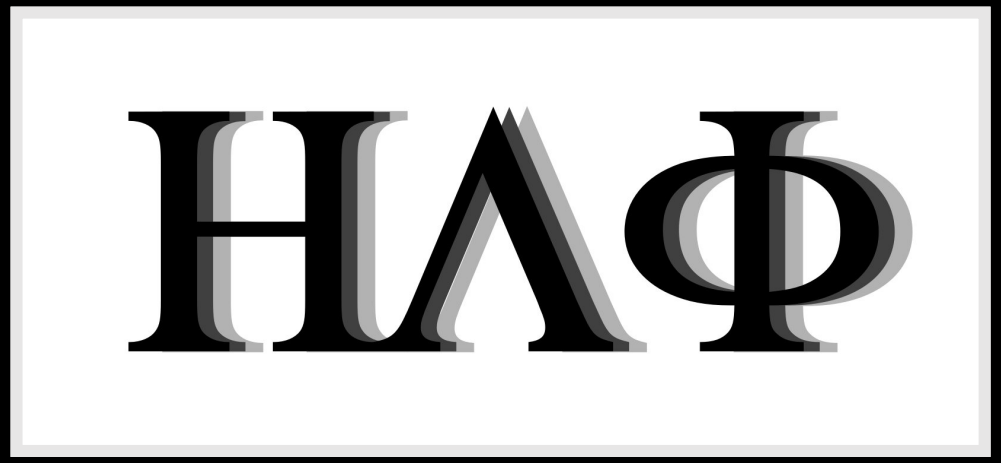
- На текущий момент существует 6 сервисов, где один **основной** сервис — HLS, два **прикладных** сервиса — HLM, HLF, три **вспомогательных** сервиса — HLT, HLE, HLL
- В описании сети «Hidden Lake» могут существовать также специфичные сервисы — **адаптеры**, именуемые как HLA. Они исполняют роль «вживления» анонимизированного трафика в инородную систему

HLS (Hidden Lake Service) — **ядро** анонимной сети.
Представляет **API** для отправления / получения
сообщений поверх анонимизирующего трафика



HLT (Hidden Lake Traffic) — **распределитель** трафика в анонимной сети. Может исполнять роль ретрансляции и хранения трафика

HLM (Hidden Lake Messenger)
— анонимный **мессенджер**,
вызывающий функции HLS



HLF (Hidden Lake Filesharer) — анонимный **файлообменник**, вызывающий функции HLS

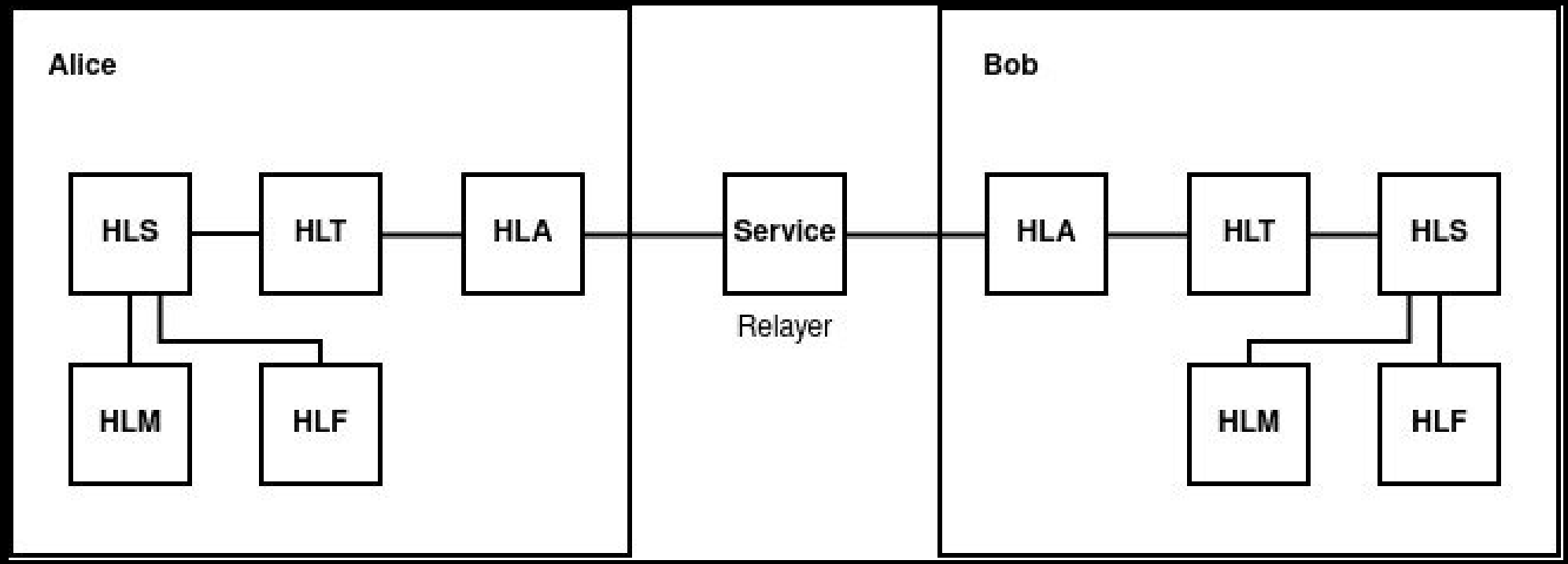
HLE (Hidden Lake Encryptor) — сервис **шифрования** и расшифрования сообщений формата **go-peer**

HLL (Hidden Lake Loader) — **скачиватель** и распределитель трафика между несколькими HLT сервисами

HLA (Hidden Lake Adapters) — **адаптеры** для создания анонимных коммуникаций в централизованных системах

Формальное описание композиции сервисов

$$Hidden\text{-}Lake = \Sigma^{n}_{i=1} APP_i \times HLS \times (HLT \times \Sigma^{m}_{j=1} HLA_j)^t$$



Возможные способы применения анонимной сети «Hidden Lake»

1. Защита локальных / корпоративных сетей от прослушивания
2. Защита военных коммуникационных узлов от прослушивания
3. Усиление безопасности уже готовых / сформированных систем
4. Использование существующей платформы для создания собственных приложений

Ссылки

- Проект go-peer

https://github.com/number571/go-peer

- Документация

https://github.com/number571/go-peer/tree/master/docs

- Директория Hidden Lake

https://github.com/number571/go-peer/tree/master/cmd/hidden_lake