PowerShell, Графы + профит

или как находить зависимости между всем и всем

DevOps или Системный инженер, вот в чем вопрос

- Менеджер: Заказчику нужен DevOps инженер ...
- DevOps: Мы!
- Менеджер: ... чтобы построить гибридное облако и перенести приложения в публичную его часть из приватной части
- DevOps: а нет, не мы

Потому я ...

- Системный архитектор (по совместительству - инженер)

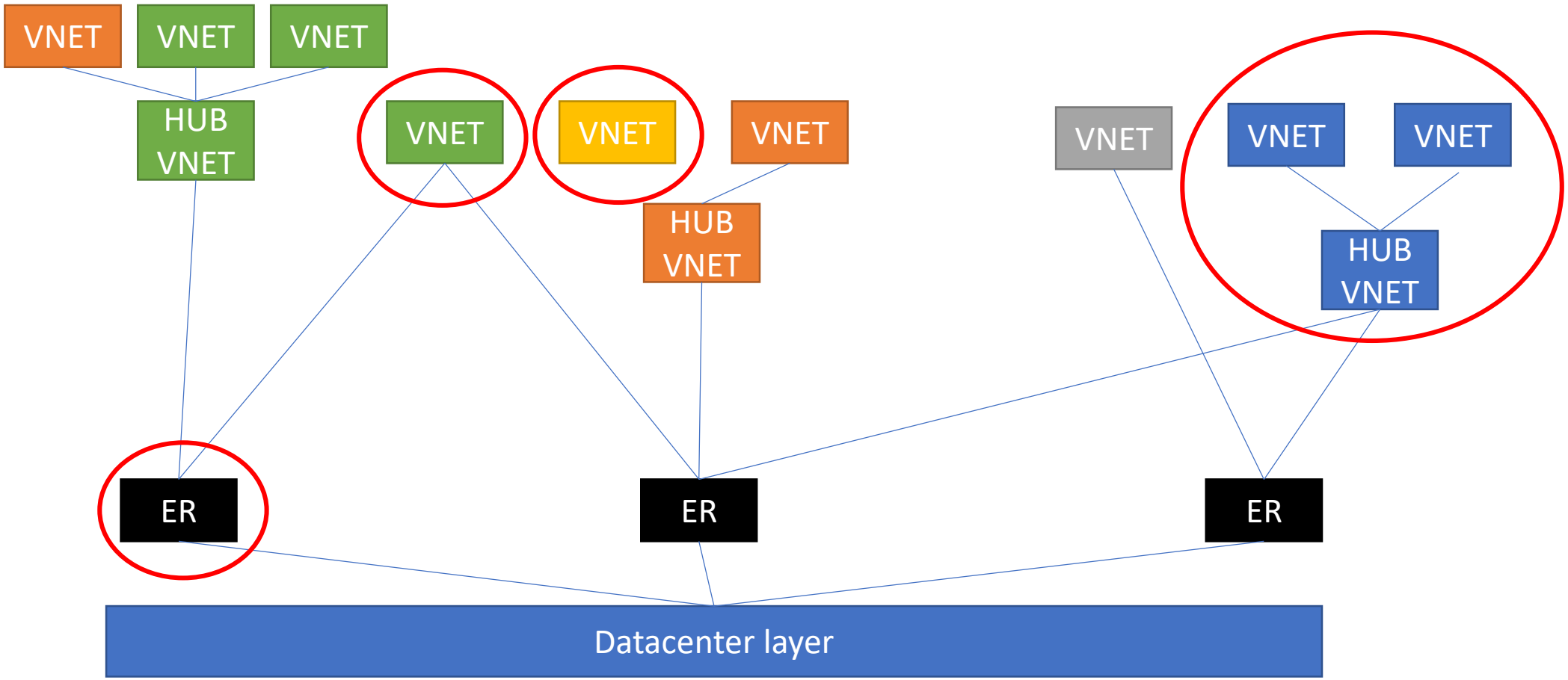
Мы занимаемся

- Переносом систем приложений оттуда сюда, ну или оттуда сюда, это как кому захочется
- В последствии поддержкой этих систем в рамках ITSM моделей заказчика
- Построением гибридных облаков, это сейчас модно
- Борьбой со сложными техническими проблемами в инфраструктурах заказчика
- Планированием и построением всякого рода платформ на основе стандартных решений в корпоративном секторе
- Ну и конечно автоматизация всего и вся в рамках поддержки этих инфраструктур. Куда ж без нее

Миграция - как есть, без изменений.

- Кому мы шлем данные?
- Кто шлет нам данные?
- Кто пользуется системой?
- Вспомогательные системы, которые мы используем
- …

Управление облаками и их содержимым



Ну и всякое другое

- Нарисовать сетевую топологию заказчика, не зная о сети ничего
- Найти с кем разговаривает наша система
- Посмотреть зависимости в нашем коде
- И так далее ...

Проблема - решение

- Проблема – поиск зависимостей между компонентами системы
- Решение – собрать метаданные, обработать и представить в виде графа
  - Инструмент – PowerShell + PSQuickGraph

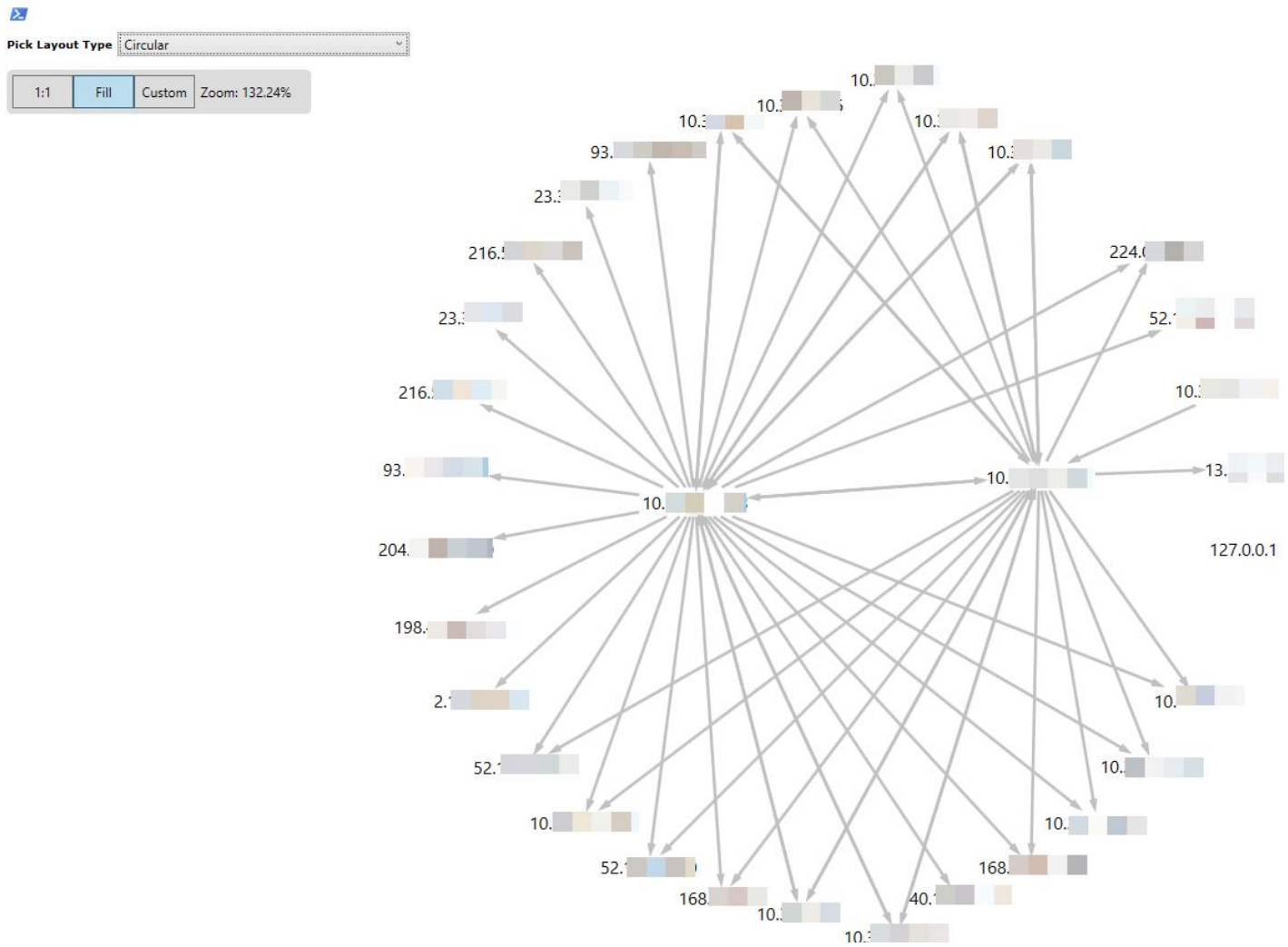
Миграция и поиск зависимостей приложений

- Данные о зависимостях можно получить из:
  - Текстовых логов firewall
  - Sysmon
  - Логи DNS
- Парсим логи
- Заталкиваем данные в граф

```
$g = new-graph -Type BidirectionalGraph

$log | ? {$_.srcip -and $_.dstip} | % {
    Add-Edge -From $_.srcip -To $_.dstip -Graph $g | out-null}
```

Получаем граф взаимодействий!



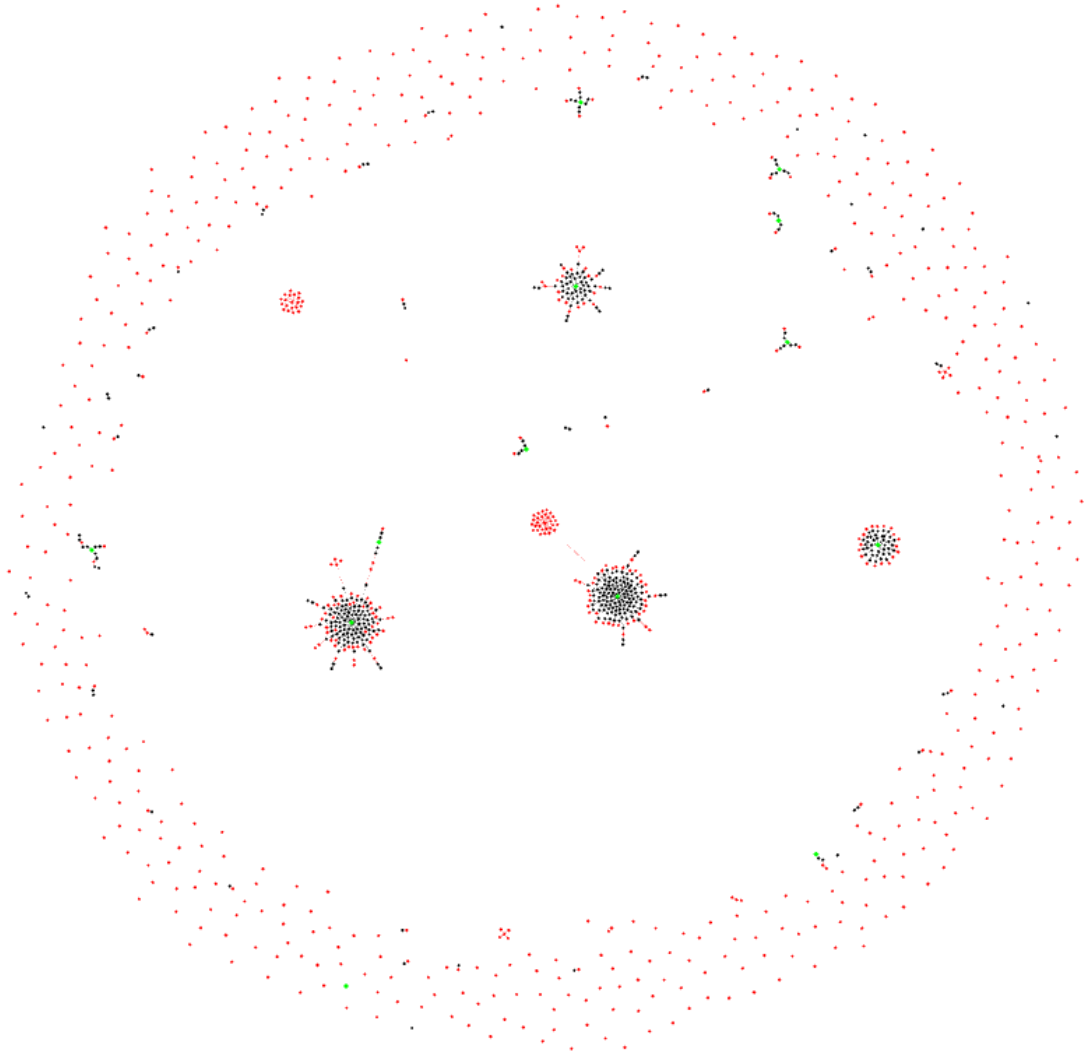
DEMO

- Simple graph
- Сетевые коммуникации
- Структура модуля

Структура объектов в облаке\governance

- Выгружаем все объекты как есть (сети, vnet peering, express route circuits …)
- Проходим раз - добавляем в граф вершины
- Проходим еще раз – добавляем дуги
- Чтобы найти все сети, поключенные к конкретному ER – вызываем алгоритм Дейкстры для всех сетей

Получаем структуру сетей в облаке



DEMO

Выводы

- Видеть связи между компонентами системы очень полезно
- Можно находить эти связи автоматически или полуавтоматически
- Графы помогают исследовать зависимости

Спасибо!

- Вопросы

Andrey Vernigora
https://github.com/eosfor
https://twitter.com/pwsh_guy
https://eosfor.github.io