

Росатом – Гринатом

Защищенный браузер

Контекст / описание проблемы:

После нескольких успешных АРТ-атак на компанию N руководство организации решило изолировать сегменты клиентской локальной сети от Интернета. Для безопасной работы пользователей в Сети применялась публикация веб-браузера на ферме терминальных серверов. Обмен файлами был реализован через общий сетевой ресурс, интегрированный с «песочницей» через REST API.

- По мере роста компании стало ясно, что эта концепция имеет серьезные недостатки:
- Перегрузка процессоров фермы терминальных серверов;
- Низкий уровень удобства использования;
- Высокая стоимость доступа к Интернету для пользователя. (Уровень энергопотребления терминальной фермы);
- Сложно прогнозируемый рост потребления ресурсов.

Задача:

Предложить альтернативный безопасный подход, который позволит снизить стоимость клиентского доступа к Интернету благодаря применению технологий виртуализации. Решение должно представлять собой Python-обертку вокруг платформы KVM, которая обеспечивает работу в интернете для 3000 пользователей в лесах Active Directory. Решение будет обеспечивать централизованную аутентификацию по протоколу kerberos, управление (настраиваемые политики для загрузки с сервера, загрузки на сервер, печати, использования буфера обмена, использования браузера), ведение журнала. Для пользователя рабочего места (ОС Windows 7 – Windows 10) решение должно выглядеть как «родное» приложение для его операционной системы. Пользователь должен взаимодействовать только с окном браузера. Другие окна должны быть скрыты. При каждом запуске пользователем решения, оно должно выполнять восстановление до стандартного состояния приложения, но при этом должен загружаться постоянный профиль данного пользователя (закладки, сертификаты X.509 и т. д.). Браузеры, используемые решением: Chromium или Firefox по вашему выбору.

Требования к решению:

1. Концептуальная схема с текстовым описанием.
2. Рабочий прототип (клиент, консоль управления).
3. Исходный код.

Технические требования к решению:

1. Исходный код: код Python
2. Диаграмма, описание: pdf
3. Прототип должен быть доступен удаленно.

Критерии оценки:

- Критерий: уровень безопасности
Шкала оценки: максимальный балл - 5, минимальный балл - 1
- Критерий: оригинальность
Шкала оценки: максимальный балл - 5, минимальный балл - 1
- Критерий: дешевизна
Шкала оценки: максимальный балл - 5, минимальный балл - 1
- Критерий: удобство использования
Шкала оценки: максимальный балл - 5, минимальный балл - 1
- Критерий: централизованное управление
Шкала оценки: максимальный балл - 5, минимальный балл - 1

Материалы для проработки:

<https://libvirt.org>

<https://code.x2go.org/doc/python-x2go/>