Система клиентской аналитики. Реализация модуля сбора данных.

1. Функционирование модуля user. По срабатыванию прописанного клиентом события, собираемые данные сохраняются в специально продуманном формате в память User (данные хранятся в сжатом и зашифрованном виде). Как только объем сохраненных на носителе данных начинает превышать некоторую величину, собранная информация отправляется модулю Spider (см. далее) по защищенному протоколу.
2. Функционирование модуля Spider. Модуль отвечает за сборку статистической информации от Users и организацию взаимодействия сервера с ними. Модуль Spider имеет реестр (список) всех Users, а также умеет устанавливать с ними безопасное соединение. При возникновении нового User или исчезновении старого, реестр корректируется.

2.1. Обмен данными с Users. Сбор данных осуществляется по следующему алгоритму: User устанавливает соединение -> осуществляется обмен ключами -> информация от User поступает на Spider -> осуществляется валидация (проверка) данных на корректность и целостность -> в случае успешного обмена информацией, соединение закрывается, при этом на User собранные данные удаляются.

2.2. Обмен данными с сервером. Взаимодействие между Spider и Server осуществляется по незащищенному сверхбыстрому каналу связи. Перед передачей информации, Spider выясняет у Server уровень его текущей загруженности и, в зависимости от полученной информации, принимает решение о передаче данных на сервер.

2.3. Логгирование. Каждое действие Spider отражается в log-файлах, ведутся журналы восстановления.

1. Функционирование Server. Информация поступает от Spider. Используются идеи классических реляционных СУБД.

Модуль Spider физически хранится в непосредственной близости от Server клиента, чтобы имелась возможность обеспечивать быстрый обмен данными между ними. Клиент имеет возможность создавать несколько физических модулей Spider для обеспечения отказоустойчивости системы (также предусмотрено создании виртуальных модулей Spider).

Особенности архитектуры: трехуровневая архитектура позволяет достичь высокого уровня надежности всей системы за счет высокого уровня модульности (отказ одного из модулей не критичен, поскольку не имеет места утеря данных). Совмещение безопасного (между User & Spider) и небезопасного (между Spider & Server) соединений позволяет обеспечить наилучшее соотношение безопасность - скорость. Наличие модуля Spider позволяет не только увеличивать надежность все системы в целом, но и обеспечивать управление загрузкой сервера. Архитектура позволяет осуществлять опрос устройств User достаточно редко (при наличии достаточного объема памяти на User), что дает преимущество в эффективности по сравнению с прочими типами архитектур.