SysInternals Suite - Top 5 тестировщика

Добрый день! Меня зовут Юрий и я работаю тестировщиком в компании Smartech. Для меня большая честь выступать на данном мероприятии с докладом «SysInternals - Top 5 тестировщика».

Как специалистам по контролю качества приложений и сервисов под Windows, нам часто приходится исследовать то, как объект нашего тестирования взаимодействует с операционной системой. Кроме того, невозможно быть хорошим тестировщиком и при этом не быть грамотным системным администратором. Сегодня я расскажу о наборе утилит, который позволяет нам глубже заглядывать и анализировать происходящее в Windows. Спектр задач, где этот набор может быть полезен:

- для мониторинга взаимодействия приложения с операционной системой и сетью во время функционального тестирования;
- для анализа поведения (в том числе падений) и профилирования приложения во время проведения нагрузочных и стресс-тестов;
- для решения прикладных задач, связанных с системным администрированием.

Немного истории. Изначально проект SysInternals Suite был создан Марком Русиновичем и с 1996-го по 2006-й разрабатывался как opensource. В 2006-м Microsoft купила проект и сразу закрыла доступ к коду, в котором демонстрировались нетривиальные решения с использованием Native API Windows. Впрочем, при желании в сети можно найти исходники версий утилит 2006-го года. Кроме того, из проекта сразу была удалена утилита для восстановления и сброса паролей системы. В настоящий момент в пакете 72 утилиты, из которых мы остановимся на 5 самых интересных для тестировщиков.
Каждая утилита может быть скачана с сайта Майкрософт по отдельности, либо в составе всего пакета. Кроме того, мы можем запустить любую утилиту непосредственно из интернета нажав Win+R и введя *\\live.sysinternals.com\tools\<toolname>.*

Философию, которая стоит за каждой из утилит, можно суммировать в нескольких пунктах:

- Нет необходимости в инсталляции и/или ребуте системы;
- Не оставляют за собой мусора на диске и в реестре;
- Не загружают наш компьютер и не требовательны к ресурсам;
- Просто работают;
- Каждая состоит только из одного исполняемого файла;
- Работают на всех версиях Windows вплоть до XP SP3.

ProcExp.exe

Самой популярной и используемой утилитой из пакета является мощная альтернатива встроенному в Windows диспетчеру задач. С появлением Windows 8 разрыв в возможностях несколько сократился, но ProcExp по-прежнему даёт фору по многим параметрам.

Перечислим некоторые основные возможности. Список далеко не ограничивается указанными, но в рамках 15-ти минутного доклада мы обязаны концентрироваться:

- Построение иерархического дерева процессов;
- Используется цветовой код для лёгкого понимания процессов;
- Очень точный расчёт использования процессора;
- Позволяет добавлять несколько разных мониторов (CPU, GPU, диск, сеть и др.) в трейбар;
- Позволяет обнаружить какой из процессов связан с открытым окном;
- Позволяет обнаружить какой из процессов блокирует файл или папку и закрыть соответствующий дескриптор;
- Позволяет видеть подробную информацию по конкретному процессу, включая треды, использование памяти, объекты и др.;
- Позволяет заморозить процесс, включая все его треды;
- Показывает какой из процессов потребляет больше всего CPU при наведении мыши на иконку монитора в трее или на пик в графике;
- Включает в себя возможность проверять любой процесс на VirusTotal.

Итак, рассмотрим 3 фичи утилиты, которых мы не увидим в обычном таскменеджере:

1. Подробнейшая информация о процессе открывается при нажатии на него ПКМ->Properties. Здесь у нас есть данные по самому исполняемому файлу, данные производительности с момента запуска ProcExp в режиме онлайн, график производительности с данными ЦПУ, памяти и ввода/вывода, информация о дисковых и сетевых операциях, график использования ГПУ, треды в режиме онлайн, соединения TCP/IP, данные безопасности, данные окружения и даже извлечённые из исполняемого файла strings, что может быть полезно для исследования подозрительных процессов. Если приложение разработано с использованием .NET, то появляется дополнительная закладка с соответствующей информацией.
2. Полный контроль над процессом по нажатию ПКМ, включая управление окном (показать окно, минимизировать, максимизировать и закрыть), выбор ядер процессора, на которых может работать наш процесс, выбор приоритета. Мы можем также перезапустить процесс, заморозить его, создать мини- или полный дамп, подключить к процессу дебагер и даже проверить его на VirusTotal. И конечно, убивать конкретный процесс или всё дерево процессов целиком, но это есть и в таскменеджере.
3. Ещё одной удобной фичей программы является проверка того, каким процессом заблокирована та или иная папка или файл. Для этого в главном окне мы нажимаем CTRL+F или выбираем Find->Find Handle or DLL… В открывшемся окне вводим часть пути или полный путь и имя файла и получаем вывод со всем процессами, которые могут блокировать нашу папку. Кстати, нам не обязательно убивать сам процесс, блокирующий папку. Мы можем нажать CTRL+H, это откроет список дескрипторов. Здесь мы можем, нажав правую кнопку мыши, закрыть именно тот дескриптор, который блокирует папку, не убивая процесс.

TCPView.exe

Что делать, если мы хотим видеть взаимодействие наших приложений с сетью в режиме онлайн? Здесь нам поможет утилита TCPView, обладающая, как и ProcExp, графическим интерфейсом. Фактически расширяя функционал netstat и визуализируя всю сетевую активность действующих процессов мы видим следующую информацию:

- Имя процесса;
- Протокол соединения (TCP/UDP/TCPV6/UDPV6);
- Локальный и удалённый адрес в виде доменных имён или ip-адресов;
- Состояние соединения.

Нажав на любое соединение ПКМ, мы можем также посмотреть подробную информацию whois конечной точки.

Кроме отображения информации монитор позволяет закрывать любые установленные соединения, что может быть полезно для негативного тестирования обрывов сетевых подключений.

А что если нам требуется записывать всю указанную информацию в виде лога для последующего анализа? Здесь нам поможет -

procmon.exe

Эта утилита перехватывает множество разнообразных событий в системе и может записывать их на жёсткий диск. В отличие от ProcExp, она предназначена быть пассивным сборщиком информации для последующего анализа и не обладает возможностью убивать процессы и закрывать дескрипторы. Как повествует Марк в своей книге, большинство кейсов, которые присылают ему пользователи, имеют схожий сценарий: «1. У нас возникла загадочная проблема; 2. мы запустили Procmon; 3. мы нашли причину проблемы».

События, которые Procmon позволяет мониторить:

- Обращения к файлам (Открытие, закрытие, чтение, запись);
- Обращения к реестру;
- Обращения к сети;
- Профилирование процессов (CPU, RAM);
- Профилирование потоков (CPU, Context switches).

Конечно, в системе ежесекундно создаются десятки тысяч событий, поэтому в мониторе предусмотрена гибкая система фильтрации по заданным критериям. Мы можем сузить выборку событий до конкретного процесса и/или типа событий. Все события собираются в лог и позволяют анализировать результаты работы монитора после того, как мы собрали достаточно информации о работе исследуемого приложения. Мы можем посмотреть детали каждого интересующего нас события, а также стек команд, которые предшествовали событию.

Для того, чтобы воспользоваться возможностью профилирования, нам необходимо активировать опцию в Options->Profiling Events. После этого, собрав достаточно данных во время работы приложения, которое может при этом подвергаться разнообразным нагрузкам, мы открываем в меню Tools соответствующий отчёт и видим собранные данные в виде графика или таблицы.

PsTools

Это группа консольных утилит, позволяющих получать информацию с удалённых компьютеров и управлять ими. На самом деле, это набор утилит с префиксом ps- который пришёл из Юникса, позволяющих:

- psexec - запускать процессы на удалённых компьютерах;
- psfile - показывать открытые удалёнными процессами файлы;
- psinfo - показывать информацию об удалённых системах;
- psping - измерять производительность сети;
- pskill - убивать процессы;
- pslist - получать подробный список процессов;
- psloggedon - видеть, кто подключён к компьютеру локально и удалённо;
- psservice - просматривать и контролировать сервисы;
- psshutdown - выключать или перезагружать компьютер;
- pssuspend - замораживать процесс.

Рассмотрим процесс работы с этими утилитами на примере psexec. Конечно, нам потребуется административный доступ к удалённому компьютеру. Формат запуска утилиты - ***psexec \\computer cmd -u domain\user -p password***. Если не указывать пароль, то утилита спросит его при подключении к компьютеру. Если же не указан и пользователь, то подключение будет осуществлено с текущего аккаунта Windows.

Так как эти утилиты обладают консольной природой, то будет излишним объяснять то, что мы можем использовать их в скриптах, автоматизирующих процессы тестирования наших приложений и подготавливающих среду для тестирования.

zoomlt.exe

Завершает наш Топ5 любимая утилита Марка Русиновича - Zoomlt. Профессия тестировщика требует от нас определенного перфекционизма, поэтому нам часто необходимо рассматривать небольшие элементы нашего приложения или сайта. Кроме того, для описания визуальных багов, при проведении презентаций и для совместной работы с несколькими участниками полезно бывает увеличивать и комментировать некоторые части экрана.

Утилита имеет 4 основных режима:

- Обычный zoom - Ctrl + 1;
- Режим рисования - Ctrl + 2;
- Live zoom - Ctrl + 4;
- Перерыв - Ctrl + 3;

Комбинации клавиш для выбора режима мы можем изменять в меню «утилиты», там же есть полное описание возможностей каждого из режимов, поэтому нет необходимости запоминать всё, что я сейчас буду упоминать.

Итак, в режиме обычного зума всё, что происходит на мониторе, останавливается и мы рассматриваем скриншот. Мы можем увеличивать и уменьшать масштаб приближения колёсиком мыши, выйти из режима приближения по нажатию правой кнопки мыши, или перейти в режим рисования по нажатию левой кнопки мыши.

Когда мы переходим в режим рисования у нас появляется небольшой красный крест. По умолчанию, он позволяет рисовать линии любой формы. Толщину рисования мы можем менять, зажав Ctrl и прокручивая колёсико мыши. Если же мы зажимаем Ctrl, то можем рисовать прямоугольники. Shift - прямые линии. Когда зажимаем Tab - овалы. А если зажимаем Ctrl+Shift - стрелки, причём стрелка рисуется немного непривычным по началу образом, от острия. Если нажать t, то мы можем печатать текст. Отменяем любую предыдущую форму по нажатию Ctrl+z, а чтобы очистить всё, что было нарисовано - e. Также мы можем менять цвета нажимая r(ed), g(reen), b(lue), o(range), y(ellow) и p(ink). Кроме того, мы можем полностью очистить экран и заменить его белым w(hite) или чёрным (blac)k листом. После того, как мы закончили с рисованием, можно сохранить скриншот в буфер, нажав Ctrl+c или сразу на диск, нажав Ctrl+s.

Следующий режим - Live zoom. Как следует из названия, мы не теряем управления и всё, что происходило на экране, продолжает обновляться. При этом регулировать степень зума мы можем, нажимая Ctrl+Up(Down). И конечно перейти в любой момент в режим рисования, нажав Ctrl+2.

Полезным бонусом утилиты является режим перерыва, который заменяет наш экран таймером. Нажимая Up/Down мы можем изменить время таймера, а в опциях мы можем сменить стартовое время, установленное по умолчанию, на удобное нам. Это может быть полезно в офисной среде, чтобы держать коллег информированными о времени нашего отсутствия. Думать о других - важнейший элемент командной работы.

Итак, мы познакомились с несколькими утилитами из набора SysInternals.

- ProcExp - визуализирует и позволяет управлять процессами в режиме онлайн;
- TCPView - отображает список конечных точек для всех установленных в системе соединений, расширяя функционал netstat;
- Procmon - собирает все события приложений и сервисов в виде лога, и позволяет нам анализировать и профилировать их работу;
- PsTools - даёт нам набор средств для дистанционного управления;
- ZoomIt - лупа, позволяющая создавать аннотации к любым элементам дизайна.

Конечно, для решения всего множества задач тестирования и системного администрирования, этих инструментов может быть не достаточно, но в запасе

SysInternals есть множество других утилит, познакомиться с которыми я рекомендую самостоятельно.

Литература для чтения:

1. Russinovich M. and Margosis A., “Windows SysInternals Administrator’s Reference”
2. Microsoft TechNet, http://technet.microsoft.com/en-us/sysinternals