Flow-Based Data Traffic Analysis

Innovation Matters

Scalable analysis of data networks in terms of traffic flows is important for IT management and business processes optimization. A system developed in the *Aurora* network traffic analysis and visualization project uses new techniques for collecting, storing and analyzing flow-based network traffic information. The system serves as a basis for developing innovative traffic analysis techniques, for instance on service relationship discovery and problem prediction.

Today, IT service teams are increasingly faced with complex distributed infrastructures. Understanding and controlling the resource usage in these infrastructures is important for successful IT management. A key objective is to optimize the provisioning and consumption of resources such as bandwidth, storage and processor cycles. Critical overload situations as well as over-provisioning should be avoided (see Figure 1).

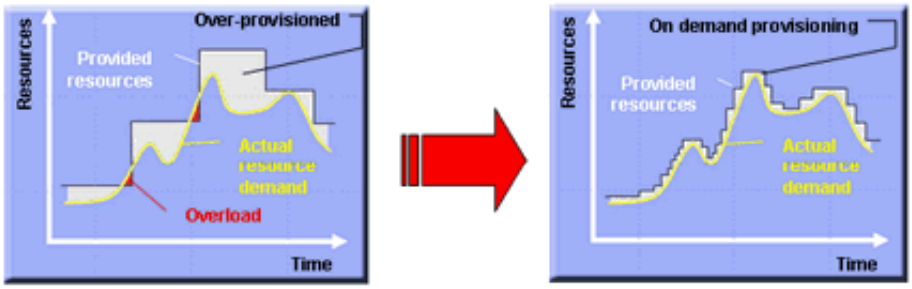

Figure 1

Resource profiling for optimized on demand provisioning

NetFlow is a widespread protocol to export traffic flow information from network routers and switches. Our work on data traffic analysis explores the great opportunities of NetFlow for analyzing and predicting resource usage in IT networks, such as data center, operator, and enterprise networks. One outcome of our work is an approach to derive direct and indirect server relationships from NetFlow. Detecting relationships and dependencies between business processes and the underlying IT infrastructure is a key premise for enabling business-driven IT management. We developed an algorithm for relationship discovery with NetFlow and applied the discovery approach in a large production environment.

This approach is very different from other relationship discovery methods which typically equip servers and end-user devices with software agents for resources monitoring and relationship detection. However, in many organizations there is a fair number of heterogeneous devices that are brought in ad-hoc and are not instrumented accordingly, for which instrumentation is not available, or on which instrumentation has been disabled. Our relationship discovery approach is much better suited for these environments because it is based on actual network flow information exported from routers and switches which connect the servers and end-user devices.

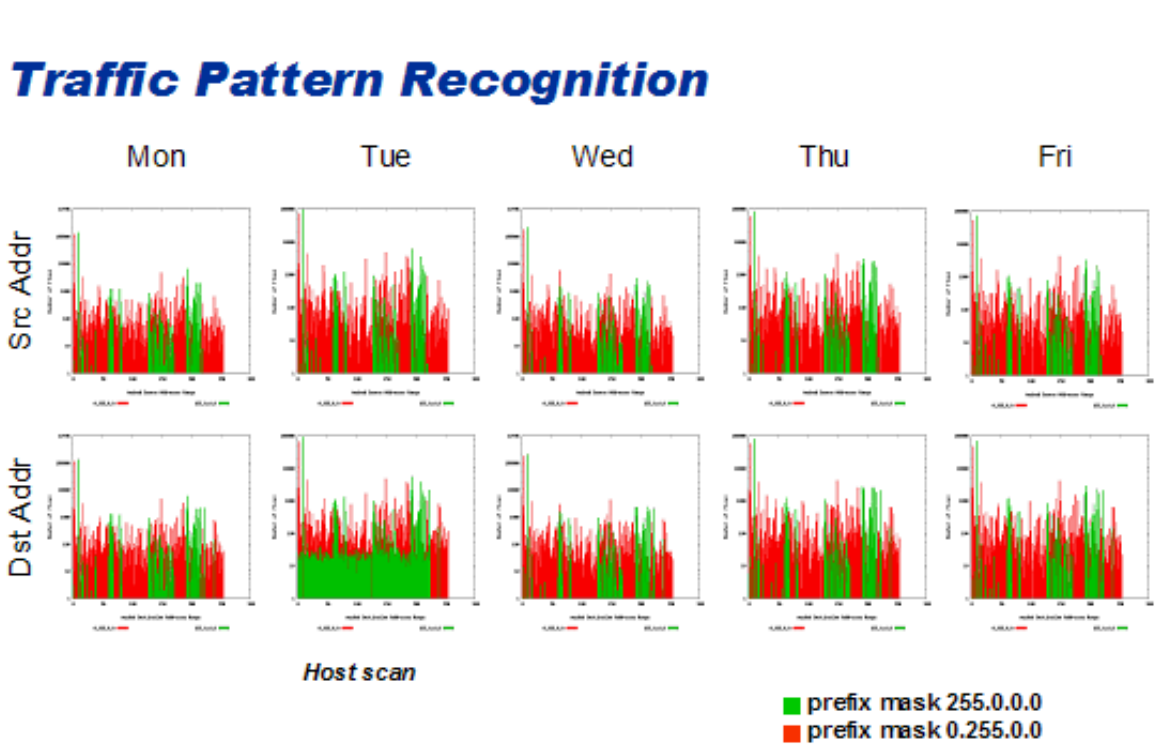

Figure 2

Average address usage pattern differs significantly on Tuesday due to host scan.

We also developed a technique for problem prediction based on behavioral workload pattern recognition. The technique was developed after observing that the emergence of malicious workloads can be much better discerned through behavioral rather than reductionistic analysis of workloads at networked devices, such as servers, storage components, routers, switches, databases, protocol gateways etc. With behavioral workload analysis, the characteristics across an entire system are observed. Parameters of individual devices are not tracked. Our experiments with behavioral workload patterns in the networking context show that the invariance during normal operation is high and the deviation during the emergence of abnormal operation is easy to detect with corresponding pattern recognition heuristics. Figure 2 shows average address usage as an example for behavioral workload pattern recognition. The usage pattern of destination addresses differs significantly on Tuesday because of a host scan caused from a computer worm. Green refers to the address usage derived with a prefix mask of 255.0.0.0 and red refers to the address usage derived with a prefix mask of 0.255.0.0.

The basis for our research a system developed in the *Aurora* network traffic analysis and visualization project (see Figure 3).. The system is equipped with a light-weight and scalable NetFlow database and analyzer. At the core of the system is a novel Aggregation Database (ADB) for time series information which provides a mechanism for efficient incremental storage of primary data values which are associated with time intervals. The database stores data values in groups of circular arrays of decreasing resolution and is, therefore, able to handle large time series data sets with fast access times and limited storage. ADB automatically assures that the array resolution of older data values is lower than the resolution of newer data values. Additionally, great care was taken with the design of ADB in order to reduce memory to disk synchronization and cache the relevant arrays in memory for fast data import and export.

Array grouping in ADB is efficient for obtaining a sorted view of related parameters. This feature is of great importance for efficiently displaying sorted lists of top protocols, top hosts, top flows, etc.

ADB has a built-in array allocation optimization which further reduces the storage requirements. If an array is updated with values of progressing intervals, no preceding array space is allocated because it will

never be needed. Furthermore, for insertion of new values, array space is only allocated in fixed chunks in order to avoid allocation of potentially unused array space. Measurements showed that in practice, these optimizations can reduce the required memory and storage allocations for only sparsely filled time series data streams up to a factor of 10. In network profiling, these optimizations are very useful when, for instance, a dynamically assigned IP address is only observed during a certain time period (e.g., a week). In this case, space in a monthly array is only allocated around the actual observation period and not for the entire month.

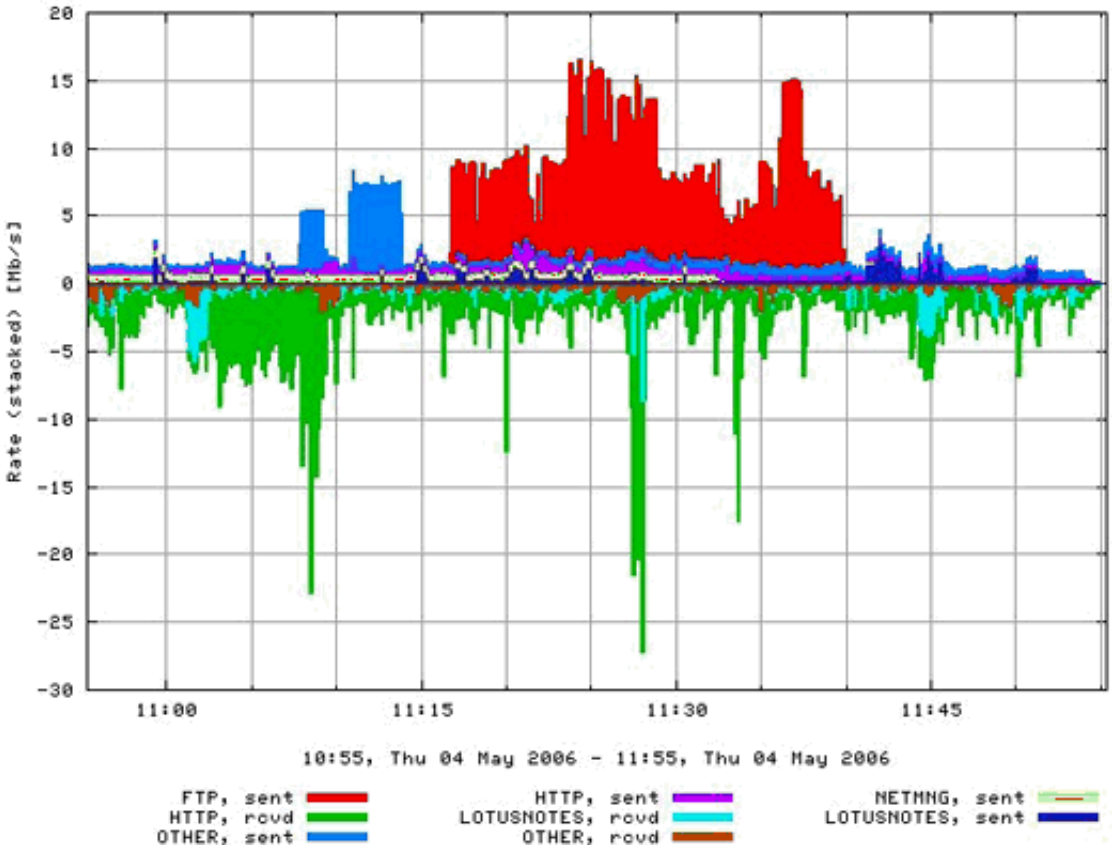

Figure 3

Bandwidth usage graph generated from aggregation database

Related Publications

Andreas Kind, Dieter Gantenbein and Hiroaki Etoh. Relationship Discovery with NetFlow to Enable Business-Driven IT Management. *IEEE / IFIP International Workshop on Business-Driven IT Management*. 2006.

News and Information
Andreas Kind, Paul Hurley and Jeroen Massar: A Light-Weight and Scalable Network Profiling System, *ERCIM News*, No. 60, January 2005

Rate this article