MONITORING OF OVERLAY NETWORKS WITH VIRTUAL RESOURCES

Jiří Navrátil, Tomáš Košnar, Jan Furman, Tomáš Mrázek, Vojtěch Krmíček
CESNET, Zikova 4, 16000 Prague 6, Czech Republic
email: jiri@cesnet.cz

Abstract
Overlay networks with virtual resources need special forms of monitoring. In the paper we describe the development of a monitoring system for the Federated E-infrastructure Dedicated to European Researchers (FEDERICA) network. We describe the whole concept and the possibility of getting basic Input/Output (I/O) statistics from different layers of the network. From the monitoring point of view, the main problem of virtual systems is multidimensionality and variability. New ways have to be found to get basic information about the load and utilization of individual resources in this type of network, whose structure is randomly changing according to the demands of its users. The VMware ESX Server-3i hypervisor is elementary and it was originally intended for small, simple systems. The low-budget experimental phase of the FEDERICA project selected it as a basic tool for the virtualization environment, but this resulted in several problems in the monitoring of resources located in the virtual nodes running in this network. It required the replacement of standard, Simple Network Management Protocol (SNMP) methods with proprietary tools, such as the VI Performance tool, RCLI from VMware and also necessitated the monitoring of the network via external devices such as Flowmon probes. In the paper we describe the actual status of our system and present some results from the monitoring of real traffic in the FEDERICA network.

Keywords
Virtualization, VMware, SNMP, Performance, Netflows, Flowmon, FEDERICA, PlanetLab, RRDTool, Nagios

1. Introduction

1.1 FEDERICA: network structure and basic principles
FEDERICA is an experimental network dedicated to the study of how virtual and traditional routers and other networking devices can collaborate in a wide-area network. It is designed as a playground for network designers and engineers, where they can test their new ideas and concepts. The monitoring system is one of its very important parts. FEDERICA is designed as a multidimensional network with a general design and is based on a slicing principle and on dedicated lines between physical nodes. The slicing principle achieved its big popularity in PlanetLab. Slicing can be described briefly as the set of parallel networks running on the set of physical nodes using the same physical connections. Compared to the Planetlab network, connected via a shared IP network, the
FEDERICA network uses individual Vpaths for each “slice”. In that sense, the virtualization process is more complex because, in FEDERICA, we play with two worlds of virtualization, the world of networking and the world of computers. FEDERICA was planned as a multivendor network of physical nodes (PN) linked with dedicated physical links (PL) in a pan-European frame. One physical node can be used in many networking applications; therefore the PN is split into several virtual nodes (VN). Each of the VNs acts as an independent computing entity which can run one or more instances of different application software. Each VN runs in the PN as an independent virtual machine (VM). Connection between VNs is made via virtual paths (VP). VPs are created within the physical networking infrastructure through different techniques. All VNs dedicated to the same purpose, together with the corresponding VPs connecting them, represent the virtual infrastructure of a slice. The slice is a complex working unit (container) of all resources and corresponding users. “Corresponding users” denotes the group of users related to a particular independent experiment or application. The slice contains all networking resources (virtual routers, virtual switches, Vpaths, etc.) together with all other resources needed for running complex applications, such as, different servers and client machines that support the purpose for which the slice was created (See Figure 1). The computing element, VM, which represents the slice in a particular VN is sometimes called “a sliver”. In Juniper Networks or Cisco Systems terminology, we are talking about “logical sub-networks or logical domains” (LD). In that sense, the slice is a similar entity but with a wider context. We intend that, in the future, FEDERICA should be a multivendor network with many different networking devices including software-based routers and switches. Currently, we are using Juniper virtual routers, MX480, for the core network and SUN machines as the virtual nodes, and running VMware as the basic virtualization platform, but we have also experimented with other systems such as Xen and OpenVZ. More details about project FEDERICA can be found in [6] or in the talks at the TERENA Networking Conference 2009 (see the streaming archive [16,17,18,19]).
1.2 Strict separation of virtual instances

As briefly described in the previous paragraph, we will have many users in the system experimenting with the running of independent networks (slices) and each user will be fully responsible for the whole slice and the application or experiment run within it. Compared to using the Linux manager, which is responsible for a standalone machine, for a FEDERICA slice manager, the problem is more complex. The slice is always running on more than one node and in each node, the VM representing this slice or sliver is shared with other slices running on these hosts. The strict separation of individual instances in the virtual environment in one VN does not allow the use of standard, internal Linux tools such as iostat, netstat or top for the whole node. Each slice user can use such tools individually inside his/her virtual machine, on a particular VN, but this gives him only the local results about resources allocated to his VM. The user will never get the full information about the global load of the whole system. Global data about utilization of the resources in the VN is only provided by tools which are embedded in the hypervisors or in other basic management infrastructure in the host operating system or special VM instances with rights to get information directly from the hardware. VMware offers the same possibility. Monitoring functions are included in the VMtool which can be used only by the node manager. This functionality is not an option for standard FEDERICA users who have a slice running on a particular node. It can be appropriate also for a FEDERICA network Network Operations Centre (NOC) that controls everything. Standard users can only get global information indirectly. In Planetlab (the popular worldwide overlay network (http://Planet-lab.org) the monitoring system runs on each node as an integral part and the system, providing the monitoring status of all resources to all users, permanently, via the Web. An example of such information is published at: http://summer.cs.princeton.edu/status/tabulator.cgi?table=nodes/table_planetlab1.cesnet.cz&sort=5&limit=50).

In FEDERICA, we are building something similar, called FMS – the Federica Monitoring System. In the following paragraphs, we describe its basic concept, giving three basic views: physical level-centric, node-centric and the slice-centric views as shown in Figure 1. The differences in the concept of VMware infrastructure management and the FEDERICA concept have made the use of VMware production monitoring tools practically impossible.

1.3 Characteristics of the virtual environment - ESX Server 3i

The Vnodes in FEDERICA run on a VMware ESX Server 3i. It is the next-generation hypervisor and provides a new foundation for a virtual infrastructure. This innovative architecture operates independently from any general-purpose operating system, offering improved security, increased reliability and simplified management. This compact and flexible architecture is designed for direct integration into the virtualization-optimized and certified server hardware. Functionally, the ESX Server 3i is equivalent to the ESX Server. However, the Linux-based service console has been removed, reducing the footprint to less than 32MB of memory. The functionality of the service console is replaced by remote command-line interfaces and adheres to system management standards. In the simplest implementation, the ESX Server 3i is embedded directly into the firmware of selected server models from various vendors, allowing the server to boot directly into the ESX Server 3i. (See: http://www.vmware.com for the latest information on vendors and models.) The ESX Server 3i is shipped with an SNMP management agent that is different from the one that runs in the service console of the ESX Server 3.
Currently, the ESX Server 3i SNMP agent supports only SNMP *traps*, not *gets*. This is one of the features of this system and one of the main reasons why we could not use standard tools to collect the I/O data with the SNMP tools; we had to develop our own monitoring system based on the proprietary software. There are some basic tools which can be used for such purposes. The remote command-line interfaces, RCLI, enable scripting and command-line–based configuration of the platform. They provide a library and a set of commands that allow the performance of the most important administrative tasks over an encrypted and authenticated communication channel.

2. SNMP Monitoring

Since the very beginning of the project (summer 2008) it was clear that most of the network monitoring would be performed via the SNMP, the system that appeared to be the best choice for FEDERICA. There were several candidates (MRTG, Nagios with CACTUS, etc.). Because we had to start basic monitoring very quickly, practically from the day that the first MXboxes were installed in core sites (in summer 2008), we decided on the G3 system [2], the system used for monitoring in the CESNET network. Because the G3 system was developed in CESNET, we were able to customize it relatively easily for the additional requirements (higher quantity of Vpaths and Vports). In CESNET, we monitor several thousands of ports with G3 and therefore assumed that we could also use it also FEDERICA. We installed a new machine and, within a few days, we deployed the new system for use in the FEDERICA core.

Our next step was to extend the system to all sites and all ports (physical and virtual). Each SNMP system is based on polling and it must know all of the devices that should be monitored. To provide specific views: “node-centric” and “slice-centric”, we also needed information about how the ports or the Vports in the MX boxes and in the Vnodes are allocated into the slices. The exact evidence of all resources (especially those created dynamically as virtual ones) and all their relationships is the fundamental problem for all such systems. We believe that tools that create networking substrate and individual VLANs for each slice will also provide registration of virtual devices into a central database. At this moment, is registration done in the NOC, by hand, and this affects complex monitoring.

The design of a monitoring system is usually based on the basic design of the network, its functionality and its main purpose. From a monitoring point of view, in FEDERICA we will have two basic levels – physical and virtual. However, the situation is not so simple. It is complicated by different environments and the tools which are available for this work. Using standard SNMP tools, we can easily monitor all of the networking devices connected via physical links (Physical level view) An example begins on the monitoring map (Figure 2) which represents global orientation, and continues in the sets of standard graphs, shown in following figures. A click on particular line in the map allows us to obtain full details of traffic as seen in Figures 3 and 4. Data from the graphs (in RRD form) can be also directly downloaded for utilization in another user application. Similarly we can monitor all of the Vpaths on the core routers and publish information about I/O on all child Vports that are linked to the corresponding physical port. We have tested it on several testing slices, which were created in FEDERICA for this purpose. However, a complete view of complex characteristics of a slice requires the integration of information from several resources. The first source is the VLAN source and the second must be obtained from the VMware.
3. Vnode monitoring based on VMware proprietary tools
VMware offers several ways getting information about status, utilization and the load of internal resources (CPUs, memory, discs, networks, etc.). The first one is the VMware Infrastructure client, which is a basic form of management for remote users, using the server platform ESX 3.i. The second way is via direct Secure Shell protocol (ssh) into the virtual node and the running command rsxtop (the top command for the VMware environment). The third way is to use the Perl toolkit with a Remote Command Line Interface -RCLI and the fourth is via a VMware vCenter. Nevertheless, we have to say that not all of these tools are available for the ESXi platform, nor are they easily applicable for the network management of FEDERICA.

The VMware Infrastructure client is the simplest management and monitoring tool. Theoretically, it can be used from the user’s home machine (if both networks are mutually visible). Via this tool, the user can log in to the server (the FEDERICA node) with the username that always gives him access to whole physical machine. In a default case, it is the root ‘uid’. It can be another user, but still it must be the ‘uid’ with access to the physical machine and with specific access rights. We do not expect that such a ‘uid’ would be appropriate for ordinary users who have a slice in the Vnode. This is one of the problematic points for using this VI client for our purposes. This tool is appropriate for the role of a NOC or local node administrator. We can call these “special users” or more generally as “superusers”. When a superuser logs in to the server, he will get a new, complete window with the tools with which he can control a particular node and get basic information about the status and performance in the whole node. He can also get information about individual virtual instances (slivers) for each slice (Figure 5).

![Figure 5: Virtual Machine Status](image)

In the example in Figure 5, the user finds several panels and menus. On the left-hand panel, there is a list of running VMs, and in the central panel he finds different information depending on the selected menu. In the "Summary", the user is provided with basic information about main resources: processors, memory usage, etc. The VI client offers several options for seeing more details, for example, to see strip charts of CPU utilization and I/O data (Figure 6).
This tool is probably sufficient for most small, classical VMware installations. But for FEDERICA, where we expect that structure to be capable of being changed dynamically according to user demands, where many independent users are sharing the same Vnode and their slices are spread over the network it is not very a usable tool. Even if the superusers (for example, the slice managers), want to see the whole slice, they would have to visit all nodes where the slice is located, one by one. Therefore, it is obvious that the VI client is not a very effective tool for getting information from a large set of nodes. However, the VI client allows the running of the \texttt{resxtop} command which is an equivalent to \texttt{esxtop} (an extended type of “\texttt{top}”). This tool will give full status information from all remote hosts and it offers the possibility to control and manage external nodes from one place. According our experience, it is adequate for occasional use, in the event of the problems, but not for regular monitoring.

Own Perl script based on Remote CLIs was the second tool that we tested. The RCLI is a set of library modules and simple scripts composed as a toolkit (Virtual Infrastructure Perl Toolkit) depends on Perl and a number of other libraries. Theoretically it can be installed on an MS-Windows machine or on any Linux machine. In practice, it will not work outside the FEDERICA address space because no user has direct access to the Vnodes. The Vnodes run in a private network which can be accessed via one gateway point called an “access point”. For the whole FEDERICA network, we are running it on one of our monitoring machines, which has access to all of the FEDERICA sites. The Remote CLI can be also be installed in VM on any of the existing FEDERICA nodes that is running VMware, as a fully installable package (a binary image for VMware called “appliance”).

In an ESX Server\textthirtysix that does not include a service console, the RCLI and its internal console is a basic tool for control and configuration of all such sites. The users open a command prompt, execute the command and pass in the connection parameters. If the users need to administer multiple ESX Server \textthirtysix hosts, using scripts is more appropriate than using a VI client. It is possible to use set up a few typical scenarios. Users can use a configuration file or pass in connection parameters using the command line. The RCLI was mainly designed for
management, not monitoring. Commands allow the creation, manipulation or deletion of a virtual resource but it has only limited functionality for obtaining information about I/O networking, etc. From this point of view, RCLI is not very effective tool for network monitoring.

More complex monitoring tools can be created using the special development kit (SDK). With the SDK tool, VMware users can develop new applications which go beyond the capabilities offered via the standard VI Client and RCLI and via standard Web Access tools. They can automate or streamline an administration process or run monitoring tasks associated with a complex virtual infrastructure. VI API is implemented as industry-standard Web services and hosted on VM Server systems, which include XML Schema 1.0, SOAP 1.1 and WSDL 1.1. Technically speaking, since the VI API is based on Web services, any programming or scripting language can be used that provides utilities for generating client-side stubs (proxy code) from the WSDL files of Web services. However, it is preferable to use Java or C++, since SOAP toolkits are readily available for these languages. Sample client applications for these two development languages are available as part of the VI SDK package and for some caveats regarding Java samples, and specific versions of the JDK and Axis are also available from Apache. The VMware vCenter is based on these tools and it provides a scalable and extensible platform that forms the foundation for powerful management. The tool is very high-priced, which we could not afford.

Using all these instruments for developing monitoring tools is a very complex task, beyond the scope of this project, especially if the tool is only to be used for this project, with a short life time. Therefore, we decided to build some new simple tools based on the Perl VI Toolkit, using RCLI together with links of this new code to other existing tools such as RRDTool, G3 or Nagios. These tools help us to store data and also help with the graphical interpretation of results. We have developed first preliminary versions of FMS, which can collect basic performance parameters from all installed FEDERICA Vnodes. At this moment, we can collect data, such as, the number of virtual instances, the CPU usage (average or in percent), I/O statistics from the ports and disk activity and we can present the data in tabular form or in the graphs generated from the RRD (Round Robin Database) (Figures 7 and 8). The system is still under development, especially the part that would create a complex slice-centric view. In next step, we will try to connect these tools with the Nagios. More graphs and diagrams about monitoring FEDERICA can be found in presentation slides [16].

![Figure 7: Node-centric View](image-url)
4. External Monitoring Tools

The goal for this part of FMS is to prepare monitoring tools for advanced users of the FEDERICA network, who want to perform networking experiments with protocols, with packet-routing and forwarding and with security, allowing users to work with applications that need synchronization, to follow ping-pong sequences and to analyse the behaviour of specific forms of routing. The possibility of measuring connection-quality is another application that could benefit from flow monitoring, especially where multimedia applications are utilized. To achieve this functionality, we have to work with detailed information about each packet and its appearance in the system, as well as with such issues as timing information, information about the sequencing chain of packets and routing tags. We can collect them, analyse them, create flow information and analyze the stream of flows and we also can then make detailed reports and statistical summaries about traffic types and other characteristics which are embedded in the traffic. In the past, such information was available only on LAN segments (grabbing data from a local machine running NIC in promiscuity mode) and analyzing them via TCPdump or via collecting information generated from the routers (flows) and its configurations (routing tables, etc.). Currently, we are doing this monitoring on any chosen links, via taping.

There are two basic ways to do flow monitoring for this purpose: internally in the devices which are part of the network infrastructure, or externally on special probes. Both approaches generate detailed information about “flow” (ntuples of packets with same source and destination and ports with added details about timing). Netflow streams are sent into a defined collector storing this information for future analysis. If this monitoring is done on routers, the service is provided in parallel with packet forwarding, and so, during heavy traffic, the routers have...
problems. The forwarding of packets can become slower or the netflows stream can become incomplete; sometimes the network becomes flooded with monitoring data.

FEDERICA is a network for experimentation, and for network research, and we cannot be dependent on a measurement device which is part of the environment that it measures. This is the main reason why a solution based on in-line monitoring (e.g., the router inside the network generating the NetFlow data) is not acceptable. Instead, we decided to use the technology based on tapping or mirroring the traffic and using independent probes. 

We have been studying this problem for several years and have chosen the software-driven Netflow probes FP8000 MM from InveaTEch, originally developed in CESNET. They fully support the 1Gbps traffic that will be used in FEDERICA and are compatible with other tools which we want to use for the presentation of results. It is planned that these new tools should cover three basic functionalities: discovery, statistics and tracing.

The ‘DISCOVERY’ functionality refers to discovering the virtual network structure. We are able to trace the path of each packet and flow passing through the network, e.g., if we take one particular flow, we can find out, through which node and its ports it has been transferred and which VLANs it belongs to. Similarly, we can observe which VLANs go through which ports of which routers and what data are transferred inside a particular VLAN network. In the current version of the monitoring devices, we need to perform queries for these particular flows at each monitored place and then to manually correlate where this packet has appeared and where not. A more sophisticated tool for automated querying and visualization of packet routes is planned, but not yet developed. Also, accurate time-synchronization between monitoring devices is needed for accurate timestamps in monitored data and for accurate network discovery.

‘STATISTICS’ refers to the statistics of network traffic, in particular VLAN network traffic - the VLAN tag monitoring is currently functional, but still in the experimental state of support. We need to perform more testing and measuring to provide a stable version of the VLAN monitoring functionality.

The ‘TRACING’ functionality is the functionality between a defined source and destination - our solution is able to filter out the statistical information (NetFlow data) about transferred data between defined sources and destination address from a particular network node, or it is able to get this data from all monitored nodes and manually correlate them. An advanced tool for correlation of such data is proposed, as mentioned in the discovery section.

The flow exporter used for this family of probes is able to export in various export formats - NetFlow version 5, NetFlow version 9 and IPFIX. The probe communicates with a remote collector via the IPv4 or IPv6 protocol over UDP.

The system FlowMon [3] and its implementation in the FEDERICA network will be described in a special paper, also presented at the TERENA Networking Conference 2009 [18]. Figure 9 illustrates an example of these monitoring reports, processed by NfSen.
The basic concept and effective operation of monitoring tools based on FlowMon probes and is based on the presumptions that probes will be installed in more sites and on all critical paths through which the traffic flows, minimally, in all four sites that constitute the core of the FEDERICA network.

5. Conclusion

FEDERICA as a multidimensional network designed for running independent networking experiments. The independence of individual layers/slices is achieved through special types of virtualization in all of its resources; but this virtualization means nothing more than resource-sharing. Network-sharing in FEDERICA is realized in a different way than traffic-sharing in current IP networks, where most connections between users consist of fat pipes with many points of branching and traffic-sharing is represented by a wide spectrum of the packet mix. The FEDERICA network is based on a number of virtual paths into which individual users send only their own traffic. What are the differences and the advantages of this network, compared to classical packet sharing? FEDERICA creates a higher level of independence of application, and it also opens the opportunity for better utilization of multi-bunch paths (in the future optical) between the nodes, which are now required more and more often. Today, it is hard to guess what advantage this has and what it really offers for the future. All of the traffic in FEDERICA reflecting different applications, as well as the monitoring system running in FEDERICA provide a way to discover the first answers to this question.

This work was supported by the FEDERICA project Seventh Framework Program of EU grant agreement 213107 and Research project of MSMT CR MSM6383917201 - Optical National Research Network and its New Applications. We also thank Rita Pužmanová from CESNET for her helpful comments.

References


