

AN APPROACH TO IMPLEMENTATION OF VIRTUAL NETWORK LABORATORY

Martin BIELIK, Igor GRELLNETH, Štefan GULA, Peter MARKO,
Vladimír MICHALEC, Peter PÉTI, Zuzana SCHUSTEROVÁ

Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies,
Slovak Technical University, Ilkovičova 3, 842 16 Bratislava 4, tel: +421-2-602 91 829, e-mail: grellneth@fiit.stuba.sk

ABSTRACT

We build a full network laboratory simulator that is as realistic as possible. To provide the “real” feeling, we use the Cisco router emulator called Dynamips, which is able to execute Cisco IOS for certain router platforms. By combining this emulator with host and network virtualization technologies we build whole virtual network laboratory comprising of routers, switches, PIX/ASA firewalls and simple network end-nodes on which our students could practice basic networking configuration and troubleshooting. We build information system with fair, secure and controlled access for students to this simulated laboratory.

Keywords: virtual, network, laboratory, simulation

1. INTRODUCTION

The main problem of classic network laboratories on our schools is limited student access into them. Instructors have to be in the laboratories to open rooms or to control student activities. There is also limited number of computers and network devices which students can use for their education. Because of this, the amount of students which can be educated in course heavily depends on how many physical devices are placed in network laboratory. Good network devices which we need for study are usually too expensive. There aren't many schools which can afford to buy enough of such devices.

Software alternative of physical network devices are simulators but they cannot replace real devices and all of their functions. Simulators can contain bugs which originate from software development. Emulators eliminate these problems but they also have their downsides. Emulators are demanding on computing power so user must have high-performance computer to be able to use them. It is also necessary to have an original operating system for Cisco devices called IOS.

As a solution for mentioned problems we decided to develop virtual network laboratory (VLAB) which will be accessible for students from central server via web interface for 24 hours a day, 7 days a week. VLAB is complex system which manages reservations of student accesses to allocation of resources on central server, contains user database as well as database of network topologies and last but not least contains Dynamips emulator, so students can try to configure network devices like in a real network laboratory. The only limitation of this solution is availability of system resources which can be increased by adding RAM or using fast CPU on central server. Additionally, server performance can be increased by connecting several computers into GRID.

Virtual network laboratory has a couple of advantages. It is independent on software and hardware platform from user perspective. It also executes much less calculations on local computer since all of them are carried out on a central server. The biggest advantage is realism of simulations whereas the system emulates real network devices.

2. VIRTUAL NETWORK LABORATORY

VLAB contains central server connected to the Internet. Users can connect to it and simulate existing topologies. User communicates with system through web interface. Our system consists of several parts which provide individual functions accessible for user. These functions are related to each other. Communication between individual parts of system can be seen on Fig 1.

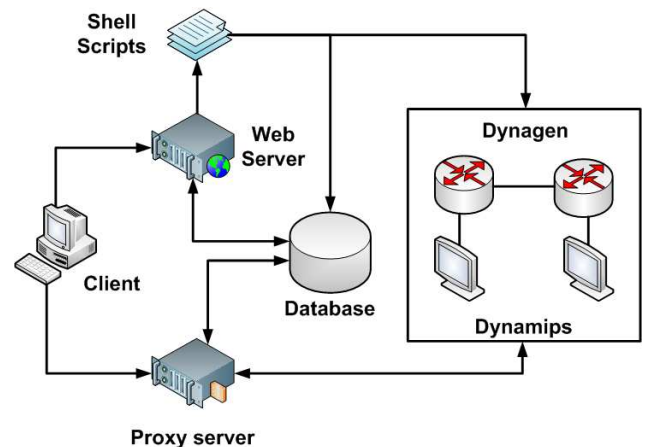


Fig. 1 Architecture of system

Architecture of system consists of these main parts:

- web server,
- PHP scripts,
- database server,
- proxy server,
- shell scripts for simulation,
- Dynamips hypervisor,
- Dynagen.

Individual parts of system are described in following chapters.

2.1. Web server

Web server is an application used to display user interface. It provides communication of user and other parts of the system. When user operation request is

received, server executes scripts written in PHP language. They generate output which server sends to user. Web server is one of the two applications, which enable access to and communication with system. It is obvious the security is critical in this part of system. We use only cipher connections through SSL protocol for security reasons. User can login to the system through secure communication to be able to change his personal data and create reservations.

2.2. PHP scripts

Good system architecture and design is crucial for system transparency and usability. Therefore it is necessary to identify each part of this system and handle it through communication interfaces. System can be decomposed into a few basic parts which can be seen on Fig. 2.

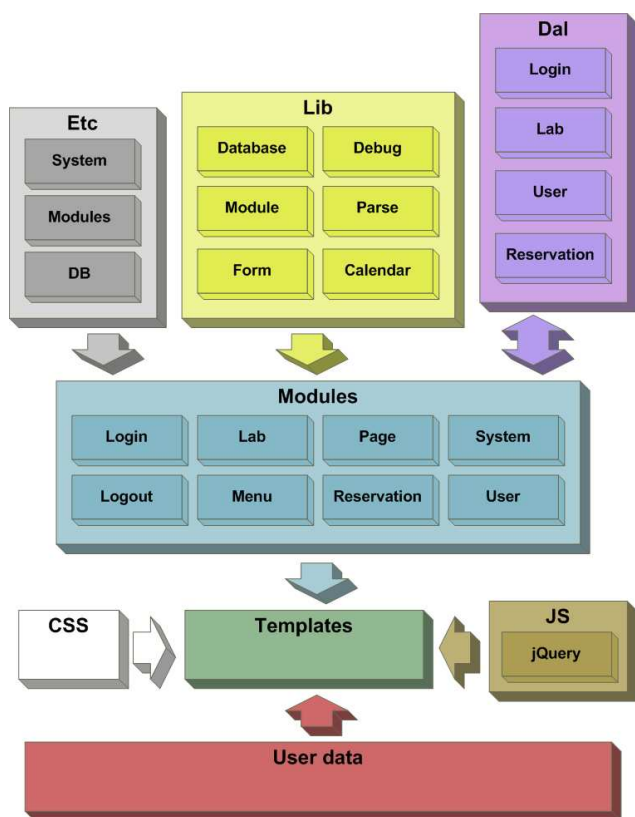


Fig. 2 Structure of the PHP scripts

We can identify three main segments in our system: system core segment, database segment, and presentation segment.

2.3. Graphviz

Graphviz is a program used for visualization of graphs. [1] Every network topology can be defined as a graph containing independent nodes connected to each other. Because of this, Graphviz is an ideal addition to our system where user inputs topology only as a configuration for Dynagen program. Topology creation is therefore much easier because in case of any topology change, system will be able to quickly change it's

representation. You can see sample output of this program in Figure. 3.

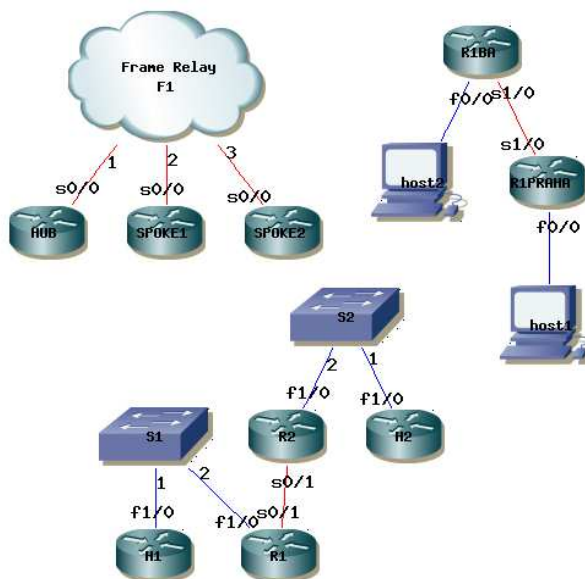


Fig. 3 Output of the Graphviz program

2.4. Shell scripts

Shell scripts are used to launch simulation from web interface. Before the simulation is launched it is necessary to change a configuration file. It is necessary to define connected workstations through VDE switch parameters and to ensure start of UML hosts. We also need to specify TCP ports through which users will be able to connect to consoles of network devices. At last, we generate XML file with parameters of the devices used in simulation. Then we start Dynamips hypervisor and insert final configuration to the Dynamips from Dynagen. In this phase devices begin to boot. Dedicated script is used to check if all devices are properly started. In case of any failure, script cancels all started processes and returns an error message to user. After successful start of all devices from topology, topology in XML structure is written to the database and script plans start of the stop script. This stop script will wait for the fixed amount of second until the reservation ends and then cancels all running processes. This way we can ensure clean finish of simulation even in case, when user logouts from the system without ending the simulation. Process identification is based on system user who started the simulation. These user accounts cannot be used to log into the system. They are used only for simulation purposes. This is why we always stop processes of user assigned to a given simulation.

2.5. Database

Database is running on same server like web interface and accepts only local connections from web server. In case of database failure, system will be inaccessible even if all other parts are running. It is because all system actions are authenticated through this database.

Database includes these tables:

- *class* – This table includes all virtual classes in our system. They are created by teachers.

- *user* – All system users are included in this table. Each of them belongs to one of three groups: administrator, teacher and student.
- *reservation* – Here we can find user reservations of system for given time. Amount of reservations is limited because system resources are limited.
- *simulation* – This table includes items needed for simulation.
- *level* – Users are divided into several groups, depending on complexity or amount of requested simulations.
- *lab* – This table includes all possible simulation topologies. There is a picture item, which include picture of topology for presentation purposes.

2.6. Proxy server

Proxy server is implemented as a simple SSH server using standard system PAM modules for authentication. User will be able to connect to device during simulation. They will use local telnet ports to do so.

Each device listens on its own port in simulation of network topology. SSH server is running on proxy server which will decide to which device will be connected. This decision is based on login. For example: Login is student_2000. Telnet starts connection on IP address 127.0.0.1 on port 2000. Finally, the whole is connection is presented to SSH server as a "shell" (Fig. 4).

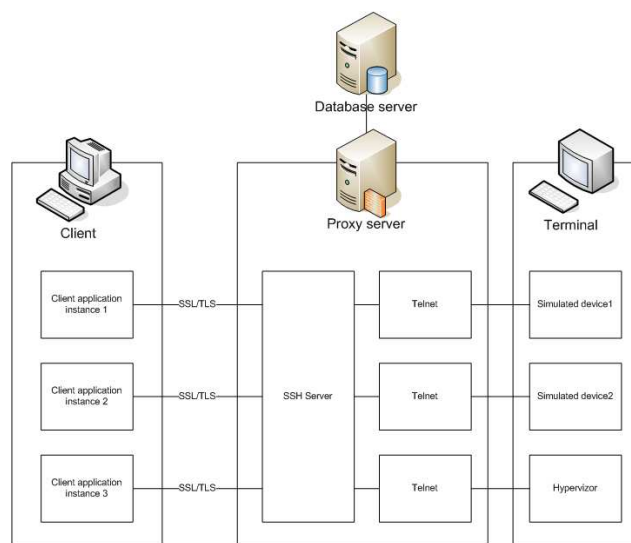


Fig. 4 Interconnection of proxy server with other devices

2.7. Dynamips

Dynamips provides emulation of processor MIPS which is able to run IOS on Cisco devices. All operations are executed by hypervisor which listens on selected TCP port. In our case, it will run on the first port of range assigned to the simulation. This process is run as an associated user too, so when user finishes simulation he stops all processes and running devices in simulation.

Communication with the hypervisor is performed only through the Dynagen or connections are created directly to the console emulated devices. [2]

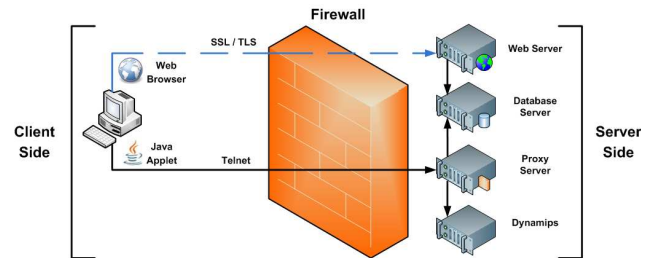


Fig. 5 Communication between the client and server

2.8. Dynagen

Dynagen processes configuration prepared by script during initialization of simulation. Dynagen ensures the hypervisor starts all necessary devices. This is the end of communication with Dynagen for actual simulation. [2]

2.9. UML host

Dynamips can emulate only devices compiled for MIPS architecture (e.g. routers). There aren't similar devices which would provide functionality of the hosts. Because of this, we used "User Mode Linux" (UML) which allows launching Linux system as common user-space application on the host system. This way, we create a full workstation as a single process, providing a standard Linux environment with minimal demands on memory which is our biggest limitation. We decided to use OpenWRT [3] distribution which is adapted to Linux devices (such as WiFi routers) from different manufacturers. This system provides a lot of standard networking tools to work with.

Script starting simulation also starts these workstations and it enters correct configuration parameters which creates connection between host and ports of connected devices. This connection is created using a VDE switch.

2.10. VDE switch

VDE stands for Virtual Distributed Ethernet. VDE switch is a system able to combine multiple applications into a virtual switch. This switch can be also set as a hub. This way the applications Dynamips, UML hosts or Qemu can communicate to each other as they can in real network. It is also possible to combine multiple VDE switches to create complex network.

VDE switch is represented as vde_switch program which creates Linux socket. Applications connected to this switch can create logical connections to it through Linux socket. Vde_switch creates only control Linux socket through which it communicates with application. This includes creation of data socket for transport of network data.

Each simulation needs to launch its own VDE switch with own control socket.

2.11. Switch

During topology selection, user can use manageable switch which will be marked differently than the classic switch. This is critical because we need to correctly identify standard switch and switch which is emulated by

router with extended switching card. System then generates necessary configuration files and when user start simulation, this router is set to act as a real switch.

2.12. PIX Firewall

Emulated PIX is created the same way as a router. This is accomplished by generating topology file for Dynagen. If Dynagen finds running Pemu server, it creates instance of virtual PIX. PIX interconnection with other devices works like interconnection of router and other devices.

Along with firewalls, in topologies oriented to network security we can initialize also simulations of some security attacks. We use hosts with Linux operating system for this purpose. These stations have Nmap program accompanied by needed scripts which start security attacks.

2.13. Users

System users are divided into three basic groups: administrator, teacher and student. Students are divided in hierarchy according to their needs in each class. This classification is used to assign part of server resources to each student. More advanced students usually need to simulate advanced topologies with many devices so this system allows them to use more resources. Teacher is needed for student's management. Teachers have higher access rights than students. They can modify student's data and they also have access to network topology management. Manager is an administrator of the system and he has highest access rights. He can work with any entity of the system. This way he is able to affect system in any state.

Detail list of the access rights in classes:

- *Student* – he can book time for simulation, he can examine offers of the topologies for class he belongs to, he can simulate chosen topology of his class, he can access already simulated topologies and finished configurations, he can change his profile and his password.
- *Teacher* – he has all rights just like student and he also can add new network topologies configurations, he can change items in the student's profile like class (only if student belongs to his class), he has access to list of the simulations of the students from his classes.
- *System Administrator* – he has all rights like student and teacher, he also can change system configuration, he can monitor system, generate output and statistics for all systems, he can manage all user accesses, he can import new students, manage all devices in the system, he can manage reservations and defined classes.

3. WEB INTERFACE

Because users access VLAB through secure web interface, they are not required to use specific operating system. Web interface enables login to VLAB, administration of system, registration of new users as well as simulations. Student doesn't need to install any extra

software applications. He only needs to have connection to the Internet. User has possibility to quit simulation, shut down web interface and then login from another computer and continues in simulation.

User needs to authenticate before he starts to work with the system. This way will user obtain adequate privilege level. These levels were mentioned in previous chapter.

3.1. User administration

User with administrator access rights can administrate user databases. He can also edit or delete individual user profiles from list of all users which you can see on Fig 6.

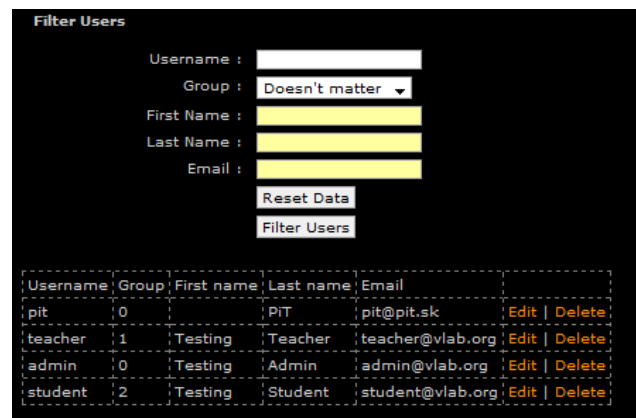


Fig. 6 List of all users

Administrator is also able to add new user profiles from this screen. System saves new user entry into user database after administrator fills in registration form. From this moment, new account is activated and user can login into VLAB.

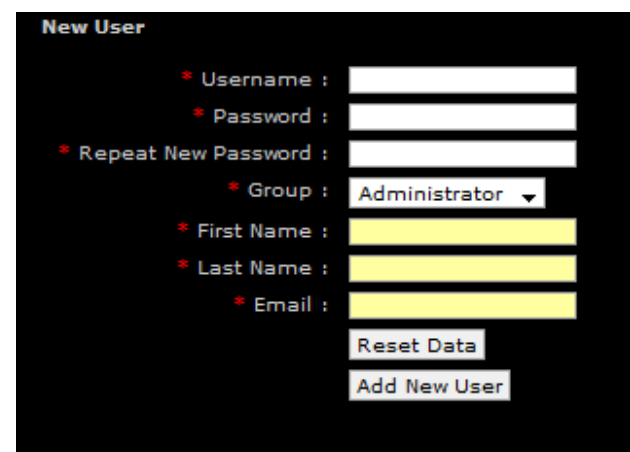


Fig. 7 Create new user

Administrator can change all user profiles. Teachers may change student profiles from their classes. If teacher needs to enable upper level labs to more skilled student, he can edit that student's profile (Fig 8).

Fig. 8 Edit user profile

Only administrator can delete user profiles (Fig 9).

Fig. 9 Delete user

3.2. Level administration

Level administration is very important from resource allocation view. A big amount of server memory is consumed by particular devices during every simulation. This is why we have to limit amount of devices which user can use in simulations. We solved this problem by implementing different user levels in the system. These levels define amount of resources needed for each reservation. Each lab exercise is assigned to level, according to the number and type of devices it contains. Each user is classified into one of these levels. We can compare levels with semesters in Cisco academy which indicate demands of students. Each level has specified maximal length of reservation duration and maximal numbers of reservations which user can create before simulation (Fig. 10).

Fig. 10 Create new level

When student creates reservation, system checks his level and shows him only times when the system is able to simulate maximum number of devices in given level. User is also able to choose simulation time with less devices but system will allocate enough resources even for more demanding simulations.

Each level has its own cost. Levels can be identified this way. When student makes a reservation, system allows him to choose only from simulations with similar or lower level. Cost can be changed in level view.

Name	Max. reservation count	Max. router count	Max. time window	
ccna1	2	2	2	Edit Delete Up Down
ccna2	3	3	4	Edit Delete Up Down
ccna3	4	4	4	Edit Delete Up Down
ccna4	4	4	6	Edit Delete Up Down

Fig. 11 Show list of levels

We can edit or delete levels from this part of web interface as you can see in Fig. 12 and Fig. 13.

Fig. 12 Edit level

Fig. 13 Delete level

3.3. Lab administration

Here we can include all functions related to lab exercises in VLAB - for example adding new labs (Fig. 15), editing (Fig. 14) or deleting existing lab exercises. Each lab contains configuration for Dynagen. These configurations are used as a specification of used network devices and connections between them. When we want to change topology, it is necessary to recalculate number of network devices and resource allocation and then assign lab to one of existing levels. Finally program Graphviz generates representation of such network topology.

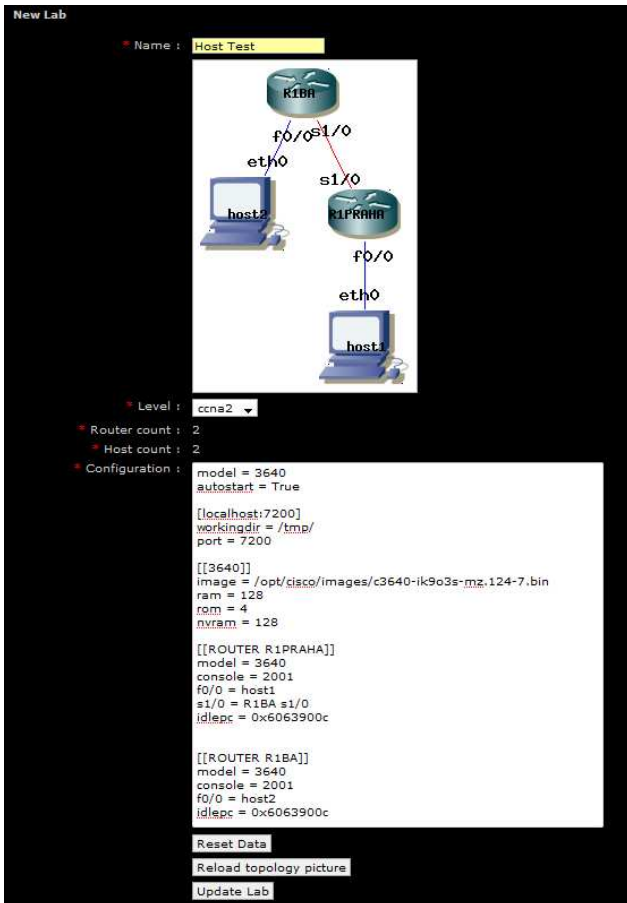


Fig. 14 Edit existing lab



Fig. 15 Create new lab

We will prepare adequate number of lab exercises in different levels so that students can try all lab exercises from Cisco academy CCNA and CCNP programs. Additionally, we create security oriented topologies which will enable simulations of different kinds of security attacks.

3.4. Reservations

Reservation is time in which user has allocated resources of server for simulation. They depend on user level. User has to be authenticated before he will be able to reserve this time. After successful authentication student can see his own reservations. If user hasn't any reservations yet, he must create a new one. Student chooses day in calendar and then system calculates free hours for simulation based on student's level (Fig. 16). Finally, user must confirm new reservation (Fig. 17).

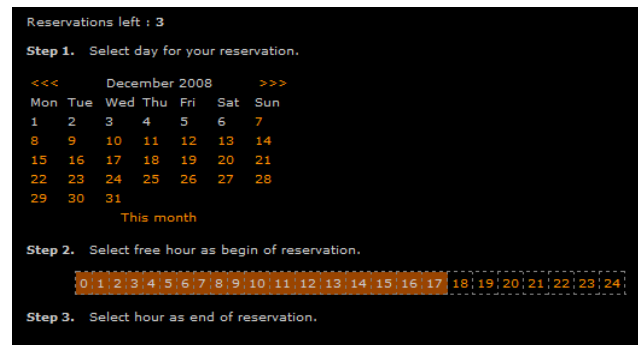


Fig. 16 Create new reservation

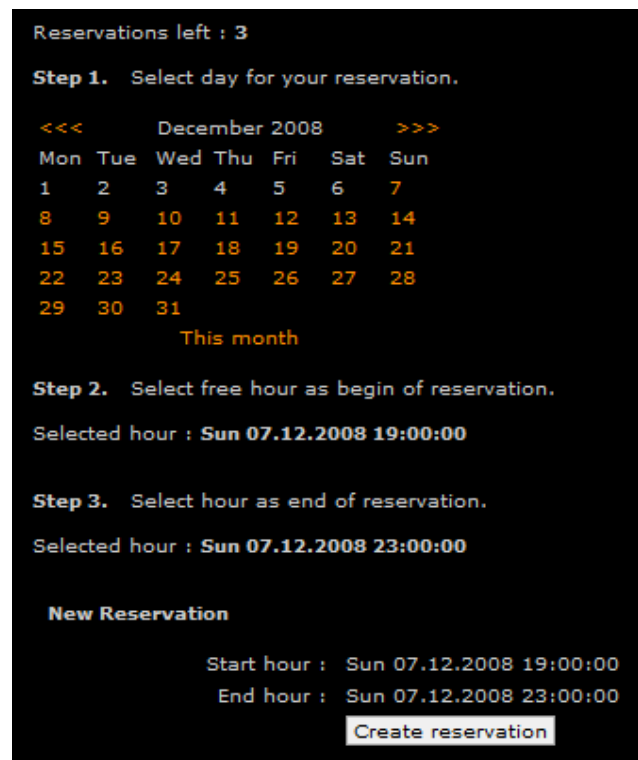


Fig. 17 Confirm new reservation

When user logs into VLAB during his reservation, he can create simulations. Student may choose one of existing labs and then start simulation. Consequently, system creates new record about simulation in database, starts all network devices in topology and shows user several ways of connecting to individual network devices (Fig 18).

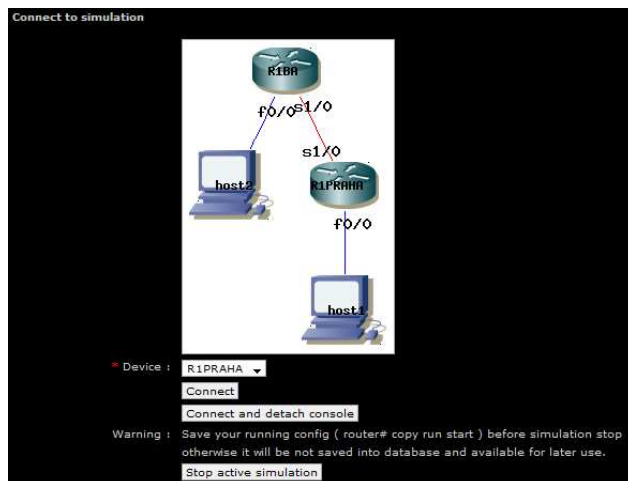


Fig. 18 Connection to console on devices

To connect to these devices, user can choose web interface, program putty, Linux console, etc.

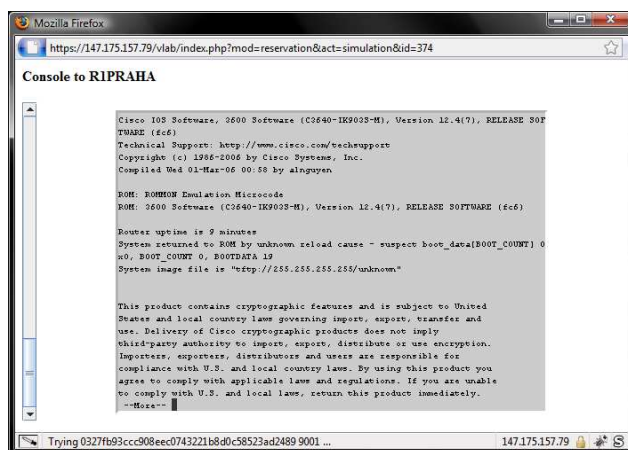


Fig. 19 Console connection to router by web interface

User can decide to finish actual simulation and choose another lab exercise. System doesn't limit students in number of simulations during one reservation but user can simulate only one lab a time. When reservation time elapses, student will be disconnected and last active simulation will be finished. System saves configuration from all devices to database during this process.

4. CONCLUSION

Nowadays, virtual network laboratory has huge potential and wide use not only in Cisco academy education. VLAB is practical teaching aid for students and teachers in school with specialization on computer networks or computer network security.

Students acquire access to network devices anywhere where they can obtain connection to the Internet, not just at school laboratories. Students can verify knowledge obtained in exercises in school laboratory, finish incomplete simulations or try many new simulations on different network topologies.

Teachers or instructors acquire overview about their students. They can see how many simulations student did, how often student uses VLAB and so on. In case of talented students, instructor can authorize simulations

from higher levels or creation of own topologies for these students.

REFERENCES

- [1] Dynamips, Dynagen, Cisco simulation, emulation. As of 20. 02. 2009. <http://www.dynamips.cn/>
- [2] Graphviz - Graph Visualization Software. As of 20. 02. 2009. <http://www.graphviz.org/>
- [3] OpenWRT. As of 20. 02. 2009. <http://openwrt.org/>

BIOGRAPHIES

Martin Bielík is student of Slovak Technical University in Bratislava. He is employed in private sector. He is active in the field of computer administration and company networking.

Igor Grelneš obtained Master Degree in Electrical Engineering from Slovak Technical University in 1993. He wrote his thesis concerning AFC System (Automatic Frequency Control) for high frequency variable oscillator. In 1994-1995, during his military duty at State Department of Environment he worked as a technician doing computer network support, troubleshooting and upgrade of all PC facilities. In 1995-1996 he was a personal assistant of the network supervisor, member of the technical staff, at the STC - Student Technology Center, University of California Los Angeles. He provided technical support for network configuration, hardware and software installation. In 2002 he has obtained his PhD degree under supervision of Professor Ing. Peter Farkaš, DrSc. In this thesis we describe two new hybrid ARQ strategies, adaptive hybrid ARQ type I for long fading channels and a hybrid ARQ scheme for reliable transport services. Further on we analyze their throughput performance and also compare it with their experimental simulation analysis. In addition there is an overview of methods for controlling transmission errors focused on hybrid ARQ strategies based on linear block codes in channel conditions with independent and dependent errors. Since January 2003 he works at Faculty of Informatics and Information Technologies in Bratislava.

Štefan Gula is student at Slovak Technical University in Bratislava. He works as instructor at RCNA FIIT STU.

Peter Péti is student at Slovak Technical University in Bratislava. He worked in Tempest where he created programs for Linux.

Peter Marko is student at Slovak Technical University in Bratislava. He is interested in network security.

Vladimír Michalec is student at Slovak Technical University in Bratislava. He works as instructor at RCNA FIIT STU.

Zuzana Schusterová is student at Slovak Technical University in Bratislava. She works as a network administrator at campus in Bratislava.