Security Risk Assessment of TeamViewer Application

Stjepan Groš
Faculty of Electrical and Computing Engineering
University of Zagreb
Unska bb, 10000 Zagreb, Croatia
E-Mail: stjepan.gros@fer.hr

Abstract. *Companies of almost any size use in their IT departments some sort of a remote access solution to help their employees. The most prominent of such solutions is a free VNC tool. Still, this tool has some serious security drawbacks, apart from being of less than satisfactory efficiency. This prompted a need for change and one solution that was evaluated was TeamViewer. Part of the evaluation process was security assessment in order to define minimal security settings for safe and secure use of the application. This paper presents this security evaluations, tries to give a more general recipe for doing such reviews, and also discusses shortcoming of this review. One novelty of this security assessment is the use of CVSS scoring system to rate threats.*

Keywords. remote support, TeamViewer, information security, risk analysis, CVSSv2

1. Introduction

Today's support to company employees by IT staff is unthinkable without some sort of a remote access solution like VNC [1] or its many variants. Yet, the installation and use of such software opens up new possibilities for attackers to gain unauthorized access to some information asset or to do some other harm to the organization. So, it is not the question weather this kind of a software will be used, but which one of the available solutions will be selected, and how the selected product will be configured in order to minimize introduced security risk.

In this paper we present security risk assessment for the application TeamViewer [2]. This is very popular application that is used for remote access in many companies and by many individuals that allows full control of a remote computer. While many features are on one hand very good, there is also other side of the coin. Many of the features are unnecessary and problematic from a security standpoint view, with the most dangerous one being the possibility to completely bypass firewall control.

It is for this reason that the security assessment of this application prior to its deployment has be done and because there was no satisfactory analysis available on the Internet, this document was written to fill this void.

Security assessment of an application should be methodologically performed. So we first present a general view of how this should be done in the Section 2. Then, using this methodology we analyze TeamViewer application in Section 3. Discussions of problems, improvements and different approaches are given in Section 4. The paper ends with conclusions in Section 5 and bibliography.

2. Principles of an application security risk assessment

In this section we try to define general methodology for security risk assessment of network applications like TeamViewer. This methodology is loosely based on NIST's risk management guide [3]. There are differences because we are evaluating a single application and not the complete information system.

2.1. Application identification and built-in controls

Today's applications are quite complex with respect to their features as well as configuration possibilities they offer. Thus it is not possible, nor it is necessary, to analyse all the possibilities. The best approach thus, would be to select some initial setup and gradually refine this

setup until satisfactory security configuration is achieved. Obviously this initial setup, as well as number of steps in the refinement, depends on the experience and knowledge of the person doing the assessment.

Security risk assessment should start with the identification of all the components that are involved in the application use as well as the security controls that the application possesses. For example, for the application that has one component installed on a server and other on a client there are two components or entities. On the other hand, a desktop application that reads and writes only local data there is only a single component, the desktop application itself. An example of security controls that majority of application do have is:

- *logging*
- *authentication*
- *network protocol and encryption*

2.2. Threat sources and goals

After the application identification, or in parallel with it, security threats should be enumerated. We have the following threat sources:

- Attackers of different skills, motivations and available resources.
- Employees.
- Malicious code.

Two things should be noted about the threat sources:

1) There is no nature or similar threat sources in a list. This is because those do not impact applications directly, but via some other means, e.g. by impacting computer on which the application is running.
2) All three attack sources can be situated on a local computer itself, on a local network, or somewhere outside the security perimeter, i.e. somewhere on the Internet.

Each threat source has some goal. The following goals should be considered when analyzing security risk of introducing new application:

- Gaining full or partial access the application itself, or to the systems which host the application.
- Information disclosure.
- Denial of service.

Note that threat list should be such that no two threats in the list fully overlap. Even if some threats do overlap, the analysis shouldn't be done twice. For example, information leakage and denial of service are completely orthogonal, that is, information leakage can not bring to denial of service, and vice versa, denial of service could not bring information leakage.

2.3. Vulnerabilities

Vulnerabilities are weaknesses that threat sources can use to achieve their goals. The following is a list of vulnerabilities that should be used for network applications like TeamViewer:

- Weak or no password protection.
- Network traffic passes in clear.
- Uneducated employee.

Note that there is vulnerability uneducated employee, which is also a threat. The reason is that uneducated employee can by itself do harm, in which case it is a threat, but can also be used by an attacker for the attacker's purpose in which case it is a mean for someone else to breach security and thus it is a vulnerability.

2.4. Performing security analysis

Finally, based on the data from application identification and threats we evaluate the possibility of threat realisation and what vulnerabilities should or do exist in order for the threat sources achieve threat goals. If there is vulnerability then we also look into controls that prevent it.

To grade severity of a certain threat realization we use modified CVSSv2 system [4]. In this modified system only the following attributes are used:

- Access Vector - which can be *local access*, *local network* or *network* in general.
- Access Complexity - which is one of *high*, *medium* or *low*.
- Impact on confidentiality, integrity and availability - which is one of *complete*, *partial* or *none*.

These attributes combine into *impact score* (impact on confidentiality, integrity and availability) and *exploitability score* (obtained from access vector and access complexity). Finally, impact score and exploitability score are

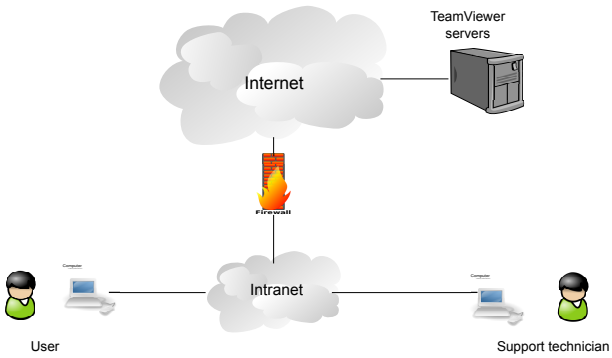

Fig. 1. TeamViewer deployment scenario used for security assessment

grouped into *CVSS base score*. To obtain values presented in this paper on-line calculator was used [5].

On a final note, special care should be taken not to confuse vulnerabilities in other applications and technologies with the application under the analysis. For example, it is certainly possible for some application to be compromised in case the underlaying OS is compromised. But this is the problem of the OS security, not of the application itself.

3. TeamViewer security assessment

The security analysis of TeamViewer application is done for a company that wishes to improve it's IT support services. This company is also very strict with respect to security and has good security controls in place. It doesn't allow outside connections or accesses without a strict control. Example of such companies are those that are in financial services, health related services, government and such.

3.1. Application identification and environment

The deployment scenario of this application is shown in the Figure 1. There is a *User* that needs a help from a *Support technician*, both of them working on their machines situated in different physical locations. In certain cases, and that's important, TeamViewer application uses *TeamViewer servers* located somewhere on the public Internet.

TeamViewer, the producer of TeamViewer application, provided a document that describes some security features of the application [6]. This document was primary source for security assessment presented in this document. In the document there is description of network protocol used by TeamViewer application, but it is only presented for the case when TeamViewer application is used in conjunction with TeamViewer servers. This mode of the operation, i.e. use of external TeamViewer servers, is unacceptable for the company for which this security evaluation is done, so that made the aforementioned reference not so useful.

Of the built-in security controls TeamViewer application has the following ones available:

- *Startup options.*
  TeamViewer can be started automatically during system boot procedure, or manually by a user. In both cases it can be configured not to allow modification of configuration settings.
- *Authentication.*
  There are three options for authentication. The first one is use of one time passwords which are generated each time the application is started. Also, fixed passwords or Windows based authentication can be used. It could be configured that the used doesn't see password, but this option is useless in case one time passwords are used.
  Password guessing protection is built into the application, even though in the available documentation there is not enough details provided.
- *Logging.*
  TeamViewer can log all connections in a single local file. This file, if properly configured, can not be altered by an unprivileged user, which is very important from the security standpoint view.
- *Network communication and Cryptographic protection*
  All communication between two TeamViewer applications is protected using strong cryptography [6]. But the details are lacking and it is hard to assess how good is the implementation. The problem is that there is not enough information concerning how long are keys valid, and how frequently they are

changed, nor is it specified how they are generated. Furthermore, one of the ways TeamViewer applications are identified is using numerical IDs but there is no description how are they generated, nor how they are protected from stealing and similar malicious activity. Finally, there is no info on details of network communication when TeamViewer servers are not used.

The following use case scenario is assumed to be the most frequently used, and this scenario will form the basis for security evaluation:

1) User phones technician because she/he has some problem.
2) Based on the user's description of the problem, technician instructs the user to start TeamViewer application.
3) Technician authenticates (using either possible authentication method).
4) During technician's work on a computer, user sees all that technician is doing.
5) After technician is done, he instructs the user to quit the application.

Finally, we make one simplification. TeamViewer offers a lot more than just remote control of a certain computer, e.g. file transfers. For lack of space, we assume that all those additional features are disabled. In more elaborate security analysis this shouldn't be ignored.

3.2. Selected security configuration

Based on short review given in the previous subsection the following configuration was selected as the most promising one:

- The application has to be installed in some system directory with properly configured file access permissions.
- The application should be manually run by the user only when technician requests so.
- The application scope should be restricted only to a local network. That is, TeamViewer servers must not be used for establishing connections.
- For authentication one time password are used, each consisting of 4 randomly generated digits.
- Application must log all connections and a copy of this log has to be stored on a remote log server or similar facility.

3.3. Security analysis

For security analysis we'll go through a list of threat goals applied to TeamViewer application, and for each threat goal we'll try to find out modified CVSSv2 value.

1) Gain full access to computer resource: By gaining full access to a computer, confidentiality, integrity and availability could be completely compromised. Thus, impact of this security risk is severe (impact subscore is 10 using CVSSv2 formulae) and proper controls should be in place in order to make this scenario impossible.

One option for an attacker, in order to gain full access to a computer on which the TeamViewer is running is to initiate communication to target computer and to pass authentication step. The following controls modulate the seriousness of this threat goal:

- The application is running in the *only local LAN mode* which means that only local network users can try to do this.
- Application is started on demand. Thus, the attacker has to guess when the application is running.
- There is one time password which the attacker doesn't know and entering wrong password delays opportunity for another try.
- The attacker can trick some employee to start application and read him a one time password. Still, in this case attacker has to be on a local network because access is restricted to local networks only. Furthermore all connections are logged.
- Attacker can hijack existing communication. But this is hard for two reasons. The first one is that he has to somehow divert communication, and the second is that the communication is encrypted.

From all this it can be concluded that restricting local network access to application greatly reduces security risk. Furthermore, to lessen security risk of some employee letting attacker in, monitoring of security logs must

be implemented also. These two measures, in addition to running TeamViewer on demand, makes exploitability complexity high, i.e. 2.5. This, in the end, gives CVSS base score of 6.5.

For the comparison, we could use unrestricted access TeamViewer application that is run upon system boot. In this case access vector is network and access complexity is medium. The reason why medium is because the attacker has to somehow find ID of the victim, which actually is somewhere between easy and hard. Finally, there is also authentication step that the attacker has to pass. All this gives new exploitability value of 8.5 and total CVSS base score of 8.5. This is substantial reduction in security risk.

2) Information disclosure: This type of threat goal, if realizes, has impact on confidentiality either complete or partial, but has no impact on integrity and availability. This gives impact subscore of 6.9.

One possibility for information disclosure is for the attacker to come into the possession of some piece of information from the computer running TeamViewer application itself, like some important system file. This could be done using included transfer protocol in TeamViewer but, as it is stated at the beginning of this section, we treat it as blocked and thus, this is not a possibility.

The attacker can also find out when some computer is powered up and/or when the technician is helping the user. This is rated as partial confidentiality impact and it can be done by simply trying to connect to the machine. If the connection is successful then TeamViewer is running. But, in order to be able to connect, the attacker has to be on a local network. Next, if the attacker successfully connects to the target machine it doesn't know if TeamViewer is run as a service or on demand. Still, with enough time, he can find out that information.

In any way, if the attacker is on the local network, access complexity of this attack is low and it doesn't require authentication. So, the CVSSv2 exploitability score is of this threat goal is 6.5, but impact subscore is 2.9 which gives CVSS base score of 3.3.

3) Denial of Service: The final threat goal we analyze is denial of service. The impact in this case is only on the availability. This impact can be such that technician can not access user's computer. Still, this is remedied by two controls. The first is that the attacker has to be on a local network. By being on the local network attacker could be easily traced meaning that this can not last for prolonged periods of time. Thus, the impact on availability is only partial. This give impact subscore 2.9.

To determine exploitability score we note that the attacker, because of the TeamViewer setup, has to be on a local network, furthermore he has to take some extra steps in order to perform this attack which grades access complexity as medium, and there is no authentication step. This gives exploitability subscore 5.5 and final CVSS base score 2.9.

4. Discussions and Future research

The presented security evaluation of the TeamViewer application gave some measure of the gains achieved using different controls. The initial configuration selected, created some common sense and experience, gave reasonably well results but also reduced the number of cycles necessary to come to satisfactory level of security as measured by CVSS scoring system.

Still, even though a good first step in a right direction, the security assessment presented in this paper suffers from a several serious drawbacks: it's not reusable, it is not complete, it is not fully correct, and there are no baselines to compare how secure it actually is.

Some drawbacks are caused by the complexity of the application. Even the moderately small application like TeamViewer has a vast number of different configuration options which makes general, and precise, security assessment almost impossible. Because of this complexity it was necessary to reduce the scope of the security assessment to a specific environment, defined in the application identification section, and to ignore all the other possibilities. Thus, this assessment is only applicable for similar environments and for different environments it has to be completely redone. This complexity of the application also makes it likely that something was overlooked which of course impacts security and makes the results obtained too

optimistic.

Another drawback of the security assessment presented in this paper is that there is no baseline to which numbers produced by CVSS scoring system can be compared and thus, they can only be used as a relative measure for this specific case. Equally problematic is the fact that there are no statistics about gains. In other words, if we introduce some control and thus we lower CVSS score, how can we judge if it is cost effective or not?

Problematic is also the fact that for the security assessment we used only the information in the available literature. Minimal number experiments of tests were done and there is much to be learned about TeamViewer application by analyzing it's behavior. Not to mention that every application has bugs because of which the application's behavior deviates from the documented and/or expected behavior. All this would certainly have impact on the final result.

In order for a single security assessment to be complete and generally usable, it has to be full in scope and parametrized. As we already said, this is very hard to achieve, even for a moderately small application like TeamViewer because of a vast number of configuration options and it should be done by someone with a very good knowledge of the application itself, preferably by the company producing application itself.

Finally, as argued elsewhere, a detail in an application might be unimportant for the security of restricted environment in which the application is analyzed, but when looked in the broader view, this detail could became crucial. For this reason the security assessment should be described in a machine readable form that will allow this assessment to be combined with the security assessment of other components so that the security of a system as a whole can be analyzed more thoroughly.

5. Conclusions

This paper presents a general security assessment of the TeamViewer application. Because the application has a large number of options it was necessary to restrict assessment for a particular use case and setup. For this use case we enumerated threat sources and threat goals and for each threat goal we estimated it's severity using a restricted subset of CVSS scoring system.

General approach to security assessment based on this specific instance for TeamViewer is also given that can be used as a template for any type of application that resembles TeamViewer. This method includes a list of suggested threat sources and threat goals, a minimal list of vulnerabilities that should be assessed, and some controls that every application have to possess.

Finally the discussion was given that identifies weaknesses of the security assessment and outlines further research that should be done in order to improve it.

Acknowledgment

This work has been carried out within project 036-0361994-1995 Universal Middleware Platform for Intelligent e-Learning Systems funded by the Ministry of Science and Technology of the Republic Croatia.

References

[1] "Virtual Network Computing," February 2011. [Online]. Available: http://www.csd.uwo.ca/staff/magi/doc/vnc/

[2] "TeamViewer - Free Remote Access and Remote Desktop Sharing over the Internet," February 2011. [Online]. Available: http://www.teamviewer.com

[3] G. Stonebumer, A. Goguen, and A. Feringa, *NIST Special Publication 800-30: Risk Management Guide for Information Technology Systems*, National Institute of Standards, U.S. Department of Commerce, July 2002. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-30/sp800-30.pdf

[4] P. Mell and K. Scarfone, "A complete guide to the common vulnerability scoring system version 2.0," June 2007. [Online]. Available: http://www.first.org/cvss/cvss-guide.html

[5] "Common Vulnerability Scoring System Version 2 Calculator," March 2011. [Online]. Available: http://nvd.nist.gov/cvss.cfm?calculator&adv&version=2

[6] T. GmbH, "Teamviewer security information," September 2010. [Online]. Available: http://www.teamviewer.com/images/pdf/Teamviewer_SecurityStatement.pdf