DOI: 10.46299/ISG.2023.MONO.TECH.1.3.3

3.3 Ensuring information security basic components of access control

Modern networks are not inaccessible objects with clearly defined security parameters. Remote access users connect from homes and public places. Visitors, employees and partners may need physical access to the internal network to do their work. But even workers are exposed to threats through Internet access, email use, and instant messaging. Therefore, understanding such a concept as "access control" is important today.

Data integrity can be protected by providing access to resources based on knowledge and needs: different types of users require different levels of access. For example, internal users may need full access, while external users may only need access to read or view information. Users should be granted access based on their job title, duties, and functions. Resources should also have different levels of classification. For example, documents should only be classified as confidential, private, public or internal use. A detailed electronic log of transactions should be maintained so that in the event of any fraud or data loss, the logs can be reviewed to determine the root cause and culprit [113].

Information security is defined as "a state of protection of the vital interests of a person, society and the state, in which harm is prevented due to: incompleteness, untimeliness and implausibility of the information used; negative information impact; negative consequences of the use of information technologies; unauthorized distribution, use and violation of the integrity, confidentiality and availability of information" in accordance with the legislation of Ukraine. In general, it is the protection of data and system assets from those who would like to misuse them. This can mean protecting against network attackers, viruses/worms, natural disasters, adverse environmental conditions, power outages, theft, vandalism or other unwanted events.

In today's computing environment, it turns out that logical assets are at least as valuable as physical assets, if not more so. In addition, it is also necessary to protect

the data of qualified employees who participate in collective operations. Files, working inventory of qualified employees are the most valuable assets, because without them it is impossible to do business at all. Copying physical and logical assets, keeping backups elsewhere is worthwhile, but without skilled people to manage and maintain data integrity, it's easy to lose everything.

When trying to preserve assets, one must consider the consequences of the security methods that are decided to be implemented. There is a well-known saying: "The only system that is reliably protected is one that is turned off, thrown into a concrete block and sealed in a lead-lined room with armed guards - and even then it is open to doubt." While of course most will believe that a system in this state can be considered reasonably safe, it is not usable and not sufficiently productive. When the level of security increases, the level of productivity decreases. The goal of security is to find a balance between protection, usability, and cost.

In addition, when protecting an asset, system, or environment, one must consider the level of security associated with the value of the object to be protected. It is possible to apply a very high level of security to each asset, while reducing performance. In any environment where a higher level of security is planned to be achieved, an estimate equal to the value of the assets must also be taken into account in order to ensure that the level of protection of their value is justified. The price of security should never exceed the value of what it protects [114].

Determining the exact point at which data can be considered secure poses a certain threat if these security systems/programs are controlled by certain authorities or entities. After all, if they are recorded in some database, then somehow someone can gain access to them, and, accordingly, to the information stored there. Therefore, there will always be new attacks to which technologies are vulnerable. If strong passwords are used, there are other approaches that an attacker can use. For example, at the moment of disconnection of the security system and the Internet, the program may be physically accessible and, accordingly, hacked.

Determining how to get into a dangerous situation is a much simpler task. Factors in this are the processes listed below:

– untimely software update;

– use of easy passwords, such as "password" or "12345678";

– downloading infected programs from the Internet;

– opening dangerous e-mail attachments from unknown senders;

– use of wireless networks without encryption, which can be controlled by anyone.

In fact, creating a perfectly secure system is quite difficult. You can point out the areas that need to be monitored and maintained in order to take measures to protect the data. The three main concepts of information security are confidentiality, integrity, and availability, commonly known as the confidentiality, integrity, and availability triad, as shown in Figure 1. The CIA triad provides us with a model that is central to the discussion of the concept of security, and typically focuses on data security [115].

Figure 1. – Triad of CIA

Privacy is a concept similar to privacy, but it is not the same. Privacy is a necessary component of privacy, which refers to the ability to protect data from those who do not have the right to view it; it is a concept that may be implemented at many levels of the process.

For example, consider the case when a person withdraws money from an ATM. The relevant person probably wants to preserve the confidentiality of the personal identification number/code (IN), which allows in combination with the bank card to withdraw funds from the ATM. In addition, the ATM owner hopes to maintain the

confidentiality of the account number, balance, and any other information necessary to notify the receiving bank. The Bank will maintain the confidentiality of ATM transactions and account balance changes after withdrawal. If privacy is breached at any point in the transaction, the results can be disappointing for the individual, the ATM owner and the bank, and this can lead to a breach in the information security field. Privacy may be compromised by the loss of a laptop containing data; the person looking over our shoulder while we type in our password; an email attachment sent to the wrong person; an attacker infiltrating our systems or similar issues.

Integrity is the ability to prevent data from being changed in an unauthorized or unwanted manner. This may mean unauthorized alteration or deletion of data or part of data, or it may mean authorized but unwanted alteration or deletion of data. In order to maintain integrity, not only is it necessary to have a means to prevent unauthorized changes to data, but it is also necessary to be able to undo changed data that has been deformed in an unauthorized way [116].

To prevent unauthorized changes, operating systems such as Windows and Linux often implement permissions/access rights that limit the actions an unauthorized user can perform on a given file. In addition, some similar systems and many applications, such as databases, may allow the undo or reverse of changes that are undesirable.

Integrity is especially important when discussing data that serves as a basis for solving other problems. If an attacker changes the data containing the results of medical tests, the wrong treatment will be prescribed, which can lead to the death of the patient.

The last phase of the KCD triad is accessibility. Availability refers to the ability to access data when needed. Loss of availability can occur at any point in the chain that allows access to data. Such problems can occur due to power loss, malfunctions in the OS or applications, network attacks, etc. When such problems are caused by a third-party factor, such as an attacker, they are usually called a DoS (Denial of Service) attack. Access control should perform four main tasks: allow access, deny access, restrict access, and revoke/revoke access. Almost all these actions are subject to a logical description [117].

Access permission - giving a certain party or parties access to a given resource. For example, if you want to give a specific user or a specific group of people access to a file or to all files in the database. It is also possible to provide access in a physical sense, giving employees access to a specific archive with a key or ID card. Denying/denying access is the opposite of granting access. When access is denied, access of the specified party to the corresponding resource is prohibited. Most access control systems should be configured to deny access by default, while allowing authorized users to use resources to the extent that has been established.

Access restriction – access to a resource, but only up to a certain point. Access restriction is especially important when using applications that may be exposed to attack-prone software environments. In a physical sense, access control restrictions can be explained using a lock in a building: suppose there are three keys: the first one opens all the doors, the second opens only a few doors, and the third opens only one, so there is a restriction.

When considering access restrictions for software, the term "sandbox" will often come up. It is used to describe the constraints that are being imposed. A sandbox is a set of resources dedicated to a program, process, or similar entity, outside of which an entity cannot operate. Sandboxes can be very useful for storing untrusted things, such as code from public websites. An example of a sandbox can be its use in a Java virtual machine (from the English JVM - Java Virtual Machine), which runs programs written in the Java programming language. The JVM is specifically designed to protect users from potentially malicious software they may download.

Access revocation is a very important point in access control. It is important that when a user is granted access to a resource, it is possible to revoke data access rights. For example, after the dismissal of an employee, it is worth revoking his access to any files of the institution where he worked: to his work e-mail account, prohibiting him from connecting to the virtual private network (from the English VPN - Virtual Private Network) of the organization, deactivating the identification card, as well as cancel other available options [118].

Two main methods are used for this: access control lists (ACLs) and capability lists (Capability List or Capabilities). There are positives and negatives to each of these, and the ways in which the four main tasks discussed earlier are accomplished will vary depending on the method chosen to implement access control. ACLs are a very common choice for implementing access control and are typically used to control access to the file systems on which operating systems are running and to control the flow of traffic on the networks to which the systems are attached. ACLs are most often discussed in the context of firewalls and routers. ACLs are built specifically for an individual system. Their structure contains the identifiers of all processes (active units) that are allowed access to the corresponding resource, and a list of operations for each process that it is allowed to perform in relation to the resource (Fig. 2).

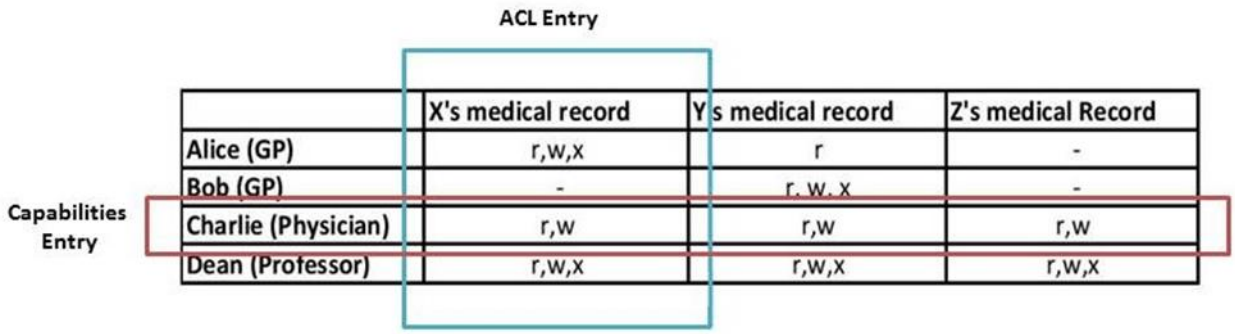

Figure 2. – Structure of ACLs and feature lists

Capability lists provide an alternative access control solution that uses a different structure than that used in ACLs. The list of capabilities lists the processes to which access is allowed, and each resource has a list of operations that this process can perform (Fig.2). Unlike capability lists, ACLs require authorization, and capability lists use a token that can be passed to one or another user [119].

In terms of network ACLs, access is controlled by identifiers used for network transactions, such as IP addresses, Media Access Control (MAC) addresses, and ports. ACLs exist at runtime in network infrastructure components such as routers, switches, device firewalls, and in software firewalls, Facebook, Google, email, or other forms of software. Access rights/permissions in network ACLs are usually binary in nature, i.e.

consist of allow and deny. When configuring an ACL, the selected identifier or identifiers are used to indicate which traffic we are accessing and to indicate whether the traffic is allowed or not.

One of the simplest forms of network-oriented ACLs is MAC address filtering. MAC addresses are, in theory, unique identifiers attached to each network interface in a system. Each network the interface has a hard-coded MAC address issued when it was created. Can usually be implemented as wireless access points. Let's use the term "access control" as an umbrella for any security concerns for accessing system resources. Within this broad definition, there are two areas of primary interest, namely authentication and authorization.

Authentication is the process of determining whether a user (or other entity) should have access to a system. The authentication problem occurs when the authentication information must pass through the network. When networks are involved, authentication is almost entirely a problem of protocol security [120].

By definition, authenticated users are allowed access to system resources. However, an authenticated user is usually not granted global access to all system resources. For example, you can only allow a privileged user (such as an administrator) to install software on the system. Then you will have to limit the actions of authenticated users, and this is the authorization field, which will be discussed below in the same subsection. Note that authentication is a binary decision – access is granted or not – while authorization is concerned with a finer-grained set of restrictions on access to various system resources. The term "access control" is often used synonymously with authorization. However, this is not the case. Access control includes both authentication and authorization.

Of course, everyone is familiar with passwords. Today, it is almost impossible to use a computer without accumulating a significant number of passwords. Even to log in to the computer, after entering the user name, the next step is to enter the password to get access to the right account if someone else is using this computer. In addition, many other things that are not commonly called "passwords" act as passwords. For example, the PIN used for an ATM card is actually a password. And if that's the case

and the password is lost or forgotten, a user-facing website can verify authentication based on a social security number, mother's maiden name, or date of birth. The problem with such passwords is that they are often not secret [121].

If they are left to their own devices (and users tend to choose easy passwords, that is, accessible information about themselves), then cracking the password will be surprisingly easy. Therefore, choosing a password should be treated responsibly, because it is possible to achieve security with the help of passwords. From a security point of view, the best solution to the password problem is to use randomly generated cryptographic keys. The problem with this approach is that people have to remember their passwords, and random bits are harder to remember. The economic factor should not be forgotten. Passwords are free, but smart cards and biometric devices require payment. In addition, they are more convenient to use: it is more convenient for an overworked system administrator to reset a password than to provide a new smart card or issue a new thumb to the user. It has already been said that cryptographic keys will solve the password problem. To understand why this is the case, you need to compare keys with passwords. On the one hand, suppose there is an attacker who comes across a 64-bit cryptographic key. Then there are possible keys, and if the key was chosen at random, then the attacker has to average the keys before he finds the correct one [122].

On the other hand, an attacker comes across a password and knows that the password has eight characters, and 256 possible choices for each character. Then there are possible passwords. At first glance, cracking such passwords may seem incredible. Unfortunately, users don't choose passwords randomly because they need to remember their passwords.

Biometrics is also used in various identification problems. That is, in identification, the goal is to identify the subject from a list of many possible subjects. This happens, for example, when a suspicious fingerprint from a crime scene is sent to a fingerprint database at a law enforcement laboratory to be compared to all the millions of fingerprint records currently in the database [123].

The problem of identification is comparing one fingerprint to many, whereas there is no such problem for authentication, because any given fingerprint is compared to

only one. For example, if an attacker wants to gain access to a user's computer and uses a biometric mouse, the captured image of the attacker's fingerprint is compared only to the stored fingerprint of the user. The problem of identification is inherently more complex and subject to much more error because of the greater number of comparisons that must be made. That is, each comparison carries a probability of error, so the more comparisons are needed, the higher the error rate [124].

There are two phases to the biometric system. First, there is a registration stage, when subjects collect their biometric information and enter it into the database. A very careful measurement of relevant physical information is usually required during this phase. Since this is a one-time job, it is acceptable if the process is slow and several measurements are required. Registration has been noted as a weakness in some field systems, as it may be difficult to obtain results comparable to those obtained in the laboratory. The second phase in the biometric system is the recognition phase. Occurs when a biometric detection system is used in practice to determine whether (for an authentication problem) a user needs to be authenticated or not. This phase should be fast, simple and accurate.