# MEMORY AUTHORIZATION IN NETWORKING: A REVIEW

**[1]Adegboye, O.J. &[2] Engr. Olaiya O. O.**
[1] Department of Computer Science. The Federal Polytechnic, Ilaro
[2]Department of Computer Engineering. The Federal Polytechnic, Ilaro
[1]Olujoba.adegboye@federalpolyilaro.edu.ng +234 70 3819 5169
[2]yinkakol@gmail.com +234 80 3812 3774

## ABSTRACT

Authorization is a security mechanism to determine access levels or user/client privileges related to system resources including files, services, computer programs, data and application features. In networking environment, memory authorization is an essential ingredient which helps the user(s) and the systems on the network generally. On a distributed system, it is often desirable to grant access without requiring a unique identity. Even when access is controlled through a combination of authentication and access control lists, the problems of maintaining the authorization data is not trivial, and often represents as much administrative burden as managing authentication credentials. This paper aimed at reviewing some different approaches used in memory authorization in networking computer systems.

**Keywords**: Smart Grid, Authentication, Authorization, Insider Threat, Security.

## INTRODUCTION

Modern and multiuser operating systems depend on effectively designed authorization processes to facilitate application deployment and management. Key factors include user type, number, credentials requiring verification and related actions and roles. For example, role-based authorization may be designated by user groups requiring specific user resource tracking privileges. Additionally, authorization may be based on an enterprise authentication mechanism, like Active Directory (AD), for seamless security policy integration. The smart grid (SG) is a future opportunistic platform for ensuring electrical power transmission and distribution in a reliable, secure, and efficient manner. However, there are many evolving challenges in the smart grid security.

Many SG security challenges have focused on protecting the system against various forms of external (outsider)cyber-attacks, including man-in-the-middle (MITM) attacks,intrusion-based attacks, malware-based attacks, denial of service (DoS) attacks, isolated attacks, and coordinated attacks.Numerous challenges arise with the integration of cyber and physical systems along with human behavior and regulatory policy. Some challenges are quite similar to those of traditional networks, but involves more complex interactions.The smart grid system has various user-roles, such as operator, vendor, engineer, administrator, etc., accessing many different types of devices in its network, such as smart meter, intelligent electronic device, etc., simultaneously. It also has more strict delay and execution time requirements.Whereas authentication and authorization are executed as two separate processes in the traditional network, executingthem as one process is needed in the smart grid to handle frequent authentications among billions of devices and dynamic user-role authorizations for a large number of users.It can also reduce the total execution time, which can help to make the system more efficient to achieve its performance requirements.Most organizations used their first Web applications to offer generally available information over the public Internet, intranets, and extranets. Successfully managing and securing corporate Web resources has become a more complex challenge as Web use has matured. Organizations that need their employees to access their intranets remotely through the Internet, or that want to automate their supply chains through extranets, should consider the security and management concerns that are unique to these situations.

To take advantage of the Internet, organizations are providing Web-based access to confidential information. With these configurations, internal and external users with varying needs and permissions should be able to access different resources maintained in the corporate intranet and users should be able to access only information for which they are authorized. Adding to the complexity of the problem, few organizations have the luxury of building their information systems from scratch. Most companies need tools that can blend new technology with their existing systems to provide security to all resources and applications accessed through the Web.Wireless sensor networks (WSNs) consist of lightweight devices with low cost, low power, and short-ranged wireless communication. The sensors can communicate with each other to form a network. In WSNs, broadcast transmission is widely used along with the maximum usage of wireless networks and their applications. Hence, it has become crucial to authenticate broadcast messages. Key management is also an active research topic in WSNs. Several key management schemes have been introduced, and their benefits are not recognized in a specific WSN application. Security services are vital for ensuring the integrity, authenticity, and confidentiality of the critical information. Therefore, the authentication mechanisms are required to support these security services and to be resilient to distinct attacks. Various authentication protocols such as key management protocols, lightweight authentication protocols, and broadcast authentication protocols are compared and analyzed for all secure transmission applications. The major goal of this survey is to compare and find out the appropriate protocol for further research.

Moreover, the comparisons between various authentication techniques are also illustrated.In this survey, various existing authentication protocols in wireless sensor networks are discussed. A list of major issues and open research challenges are compared and analyzed. Moreover, an exhaustive survey on the available protocols for authentication in the wireless sensor networks and their applications is provided. The survey also contains the major aspects of examining the protocols on the basis of quality measurement as needed for authentication mechanisms. The comparison tables are provided for decision-making on the most appropriate protocols. It fulfills the requirements of the particular application scenario.

## DISCUSSION

Authentication and authorization both play important roles in online security systems. They confirm the identity of users and then grant access to your website or application. It's vital that you understand their differences so you can determine which type of website authentication best suits your security needs.To put it simply, authentication is the process that confirms a user's identity.Traditionally, this is done through a username and password. The user enters their username, which allows the system to confirm their identity; this system relies on the fact that only the user and the site's server know the password. The website authentication process works by comparing the user's credentials with the ones on file. If a match is found, the authentication process is complete.While password authentication is the most common way to confirm a user's identity, it isn't even close to the most effective or secures method.Anyone with your credentials could access your account without your permission, and the system wouldn't stop them. Most passwords are weak, and hacking techniques can break them in less and less time.Passwords aren't the only way to authenticate your users. We'll cover two alternative methods that sites can use to verify a user's identity:

### Email Authentication:

Email authentication is a passwordless option that allows users to securely log in using just an email address. The process is very similar to signing in with a Facebook or Twitter account, but this method offers a universal approach.Aside from being inherently more secure than a password, email authentication tools like Swoop will also notify users of any suspected malicious or unusual activity.Aside from being basically and mostlymore secure than a password, email(verifying someone's identity) toolslike Swoop will also tell users of any suspected evil and cruel or unusual activity.

### Biometric Authentication:

Biometric authentication includes any type of authentication method that requires a user's biology. While this may seem like new-age technology, you're probably already using it to unlock the screen on your smartphone. Fingerprint scanning is the most well-known form of biometric authentication, but face recognition tools are an increasingly popular choice for developers.Of course, hackers have a much more difficult time replicating a users' biological characteristics, but it is important to note that these authentication processes are often less secure than you'd initially assume. Small fingerprint scanners on smartphones only record portions of your fingerprint, for instance. Multiple images of part of a fingerprint are much less secure than a single, clear image.Biometric authentication can't be changed or altered if a user's fingerprints have been compromised. While biometric authentication holds a lot of promise, it's now most useful as an additional login tool to bolster another system.

## AUTHENTICATION TECHNIQUES

Cryptography provides an easy way for the transmitter and receiver to define a subset of valid messages that the transmitter can construct and the receiver can verify.

Two types of cryptosystems are available:-
i) Private key cryptosystems
ii) Public key cryptosystems: A more traditional technique that complements the two cryptographic methods is
iii) Biometric Systems

**Private Key cryptosystems**: Symmetric encryption (also called private-key encryption or secret-key encryption) involves using the same key for encryption and decryption. Encryption involves applying an operation (an algorithm) to the data to be encrypted using the private key to make them unintelligible. The slightest algorithm (such as an exclusive OR) can make the system nearly exposed.The main disadvantage of a secret-key cryptosystem is related to the exchange of keys. Symmetric encryption is based on the exchange of secret keys.

**Public-key cryptosystems**: also known as asymmetric cryptography is a class of cryptographic algorithms which requires two separate keys, one of which is secret (or private) and one of which is public. Although different, the two parts of this key pair are mathematically linked. The public key is used to encrypt plaintext or to verify a digital signature; whereas the private key is used to decrypt cipher text or to create a digital signature. The term "asymmetric‖ from the use of different keys to perform these opposite functions, each the inverse of the other – as contrasted with conventional cryptography which relies on the same key to perform both.

**Two factor and multi factor authentication**

Adding authentication factors to the authentication process typically improves security. Strong authentication usually refers to authentication that uses at least two factors, where those factors are of different types. The distinction is important; since both username and password can be considered types of knowledge factor, basic username and password authentication could be said to use two knowledge factors to authenticate -- however, that would not be considered a form of two-factor authentication (2FA). Likewise for authentication systems that rely on "security questions," which are also "something you know," to supplement user ID and passwords.

Two-factor authentication usually depends on the knowledge factor combined with either a biometric factor or a possession factor like a security token. Multifactor authentication can include any type of authentication that depends on two or more factors, but an authentication process that uses a password plus two different types of biometric would not be considered
three-factor authentication, although if the process required a knowledge factor, a possession factor and an inherence factor, it would be. Systems that call for those three factors plus a geographic or time factor are considered examples of four-factor authentication.

Authorization and Access Control are terms often mistakenly interchanged. Authorization is the act of checking to see if a user has the proper permission to access a particular file or perform a particular action, assuming that user has successfully authenticated himself. Authorization is very much credential focused and dependent on specific rules and access control lists preset by the web application administrator(s) or data owners. Typical authorization checks involve querying for membership in a particular user group, possession of a particular clearance, or looking for that user on a resource's approved access control list, akin to a bouncer at an exclusive nightclub. Any access control mechanism is clearly dependent on effective and forge-resistant authentication controls used for authorization.

Access Control refers to the much more general way of controlling access to web resources, including restrictions based on things like the time of day, the IP address of the HTTP client browser, the domain of the HTTP client browser, the type of encryption the HTTP client can support, number of times the user has authenticated that day, the possession of any number of types of hardware/software tokens, or any other derived variables that can be extracted or calculated easily.

**Discretionary Access Control**

Discretionary Access Control (DAC) is a means of restricting access to information based on the identity of users and/or membership in certain groups. Access decisions are typically based on the authorizations granted to a user based on the credentials he presented at the time of authentication (user name, password, hardware/software token, etc.). In most typical DAC models, the owner of information or any resource is able to change its permissions at his discretion (thus the name). DAC has the drawback of the administrators not being able to centrally manage these permissions on files/information stored on the web server. A DAC access control model often exhibits one or more of the following attributes.Data Owners can transfer ownership of information to other users. Data Owners can determine the type of access given to other users (read, write, copy, etc.)Repetitive authorization failures to access the same resource or object generates an alarm and/or restricts the user's accessSpecial add-on or plug-in software required to apply to an HTTP client to prevent indiscriminant copying by users ("cutting and pasting" of information) Authentication is important because it enables organizations to keep their networks secure by permitting only authenticated users (or processes) to access its protected resources, which may include computer systems, networks, databases, websites and other network-based applications or services.Once authenticated, a user or process is usually subjected to an authorization process as well, to determine whether the authenticated entity should be permitted access to a protected resource or system. A user can be authenticated but fail to be given access to a resource if that user was not granted permission to access it.The terms authentication and authorization are often used interchangeably; while they may often be implemented together the two functions are distinct. While authentication is the process of validating the identity of a registered user before allowing access to the protected resource, authorization is the process of validating that the authenticated user has been granted permission to access the requested resources. The process by which access to those resources is restricted to a certain number of users is called access control. The authentication process always comes before the authorization process.Once a user has been authenticated, the authorization process determines what permissions they have.Permissions are what the user is able to do and see on your website or server, and without them every user would have the same abilities and access to the same Authentication and authorization keep internal accounts organized and help catch unauthorized activity before it becomes a serious threat. One of the main steps we recommend to protect against breaches is to make sure accounts only have the permissions they need.

Strong security authentication protocols prevent cybercriminals from gaining access to your accounts. Having a secure authentication method will make it more difficult for hackers to crack a users' key and gain access to their information.

## OBSERVATIONS

To protect the data in a security system, administrators should be able to, among other things, implement detailed user access privileges, select the information that can be shared internally with partners and authorities, and control how long data is kept. Logical partitions and privileges are just two mechanisms that make this possible.

### Configuring logical partitions

By configuring logical partitions, administrators determine whether one or more users can actually view specific data like recorded video. If the user is not granted access to a partition, he or she will not be able to view archived video located within that partition.The next step is to define a user's privileges. For example, although a user can view archived video, his or her privileges will determine whether he or she can export, modify, or delete that video. This ensures that recordings can only be managed by those investigators with sufficient access rights. This mitigates the risk of evidence being sent to unauthorized parties.And, to further eliminate human error, organizations can use an LDAP server, like Microsoft Active Directory, to automatically add and remove security user accounts, grant access rights, or remove users when they are no longer working with the organization. When administrators manage what their personnel can see and do, they are ensuring the security of the data transmitted and stored within their security system. This not only increases the security of the system as a whole, but it also enhances the security of other systems connected to it.Each of these authorization protocols mandates different security considerations.

These security considerations are vital to best protecting the user identities and the communications between the service providers and databases. For SAML, security is based on SAML Assertions to the Service Provider, which work with HTTP POST and encryptions.OAuth 2.0 and OpenID Connect both use secure message exchanges which tend to rely on DNS and TLS integrity. This has the bonus of easy application integration, according to Ubisecure.If the network is caught between these three authorization protocols remember that there is, in fact, a significant amount of overlap between them. For example, SAML and OpenID Connect provide both authorization and authentication in a relatively equal measure. However, there are some clear contexts in which one authorization protocol will work better than another. SAML is a good choice for browser operation, yet for application usage, OpenID Connect will be a stronger choice.

## MEMORY IMPLICATIONS USING AUTHORIZATION

A security risk for local storage is Javascript can be subject to Cross-Scripting attacks (XSS). In the early days, XSS was a result of not escaping user input, but now modern web app probably imports numerous JS libs from analytics and attribution tracking to ads and small UI elements. Local storage is global to your website domain. Thus, any javascript on the network, 3rd party lib or not, can access the same local storage. There is no sandboxing within the network. For example, your analytics lib reads and writes from the same local storage as other networks application code. In the past, there was also concern if javascript made AJAX calls in plain text even though the website itself was secured via HTTPS. This concern is less than it used to now that browsers are starting to enforce checks for mixed content. Something to still be aware of incase a browser is older or launched without enforcement.A second downside for local storage is you can't access it across multiple subdomains.

Before choosing the memory authentication techniques specific to your network, several preparatory steps can help expedite and clarify the design process;

1. Try to quantify the relative value of information to be protected in terms of Confidentiality, Sensitivity, Classification, Privacy, and Integrity related to the organization as well as the individual users. Consider the worst case financial loss that unauthorized disclosure, modification, or denial of service of the information could cause. Designing elaborate and inconvenient access controls around unclassified or non-sensitive data can be counterproductive to the ultimate goal or purpose of the web application.
2. Determine the relative interaction that data owners and creators will have within the web application. Some applications may restrict any and all creation or ownership of data to anyone but the administrative or built-in system users. Are specific roles required to further codify the interactions between different types of users and administrators?
3. Specify the process for granting and revoking user access control rights on the system, whether it be a manual process, automatic upon registration or account creation, or through an administrative front-end tool.
4. Clearly delineate the types of role driven functions the application will support. Try to determine which specific user functions should be built into the web application (logging in, viewing their information, modifying their information, sending a help request, etc.) as well as administrative functions (changing passwords, viewing any users data, performing maintenance on the application, viewing transaction logs, etc.).

## CONCLUSION

There are a plethora of accepted access control models in the information security realm. Many of these contain aspects that translate very well into the web application space, while others do not. A successful access control protection mechanism will likely combine aspects of each of the following models and should be applied not only to user management, but code and application integration of certain functions.The privileges and preferences granted for the authorized account depend on the user's permissions, which are either stored locally or on the authentication server. The settings defined for all these environment variables are set by an administrator.With the increasing number of internet-enabled devices, reliable machine authentication is crucial to enable secure communication for home automation and other internet of things applications, where almost any entity or object may be made addressable and able to exchange data over a network. It is important to realize that each access point is a potential intrusion point. Each networked device needs strong machine authentication and also, despite their normally limited activity, these devices must be configured for limited permissions access as well, to limit what can be done even if they are breached.

### Conflict of interest

Traditional authentication depends on the use of a password file, in which user IDs are stored together with hashes of the passwords associated with each user. When logging in, the password submitted by the user is hashed and compared to the value in the password file. If the two hashes match, the user is authenticated.This approach to authentication has several drawbacks, particularly for resources deployed across different systems. For one thing, attackers who are able to access to the password file for a system can use brute force attacks against the hashed passwords to extract the passwords. For another, this approach would require multiple authentications for modern applications that access resources across multiple systems.

### References

[1] Noor, A. (2008, March). Identity protection factor (IPF). In Proceedings of the 7th symposium on Identity and trust on the Internet (pp. 8-18). ACM.

[2] Khan, H. ―Comparative study of authentication techniques‖, International Journal of Video & Image Processing and Network Security Vol:10 No:04,2012

[3] Masadeh, S. R., Azzazi, A., Alqaralleh, B. A., & Ali, M. A. (2014). A NOVEL PARADIGM IN AUTHENTICATION SYSTEM USING sWIFI ENCRYPTION/DECRYPTION APPROACH. International Journal of Network Security & Its Applications, 6(1)

[4]European Commission Information Society. Internet of Things Strategic Research Roadmap, 2009.

[5]A.Rahmani et al. Smart e-Health Gateway: Bringing Intelligence to IoT-Based Ubiquitous Healthcare Systems. In CCNC'15, 2015.

[6] C. Koop et al. Future Delivery of Health Care: Cybercare. EMBM, 27(6):29–38, 2008.

[7] R. Mueller et al. Demo: A Generic Platform for Sensor Network Applications. In MASS'07, pages 1–3, 2007.

[8]W. Shen et al. Smart Border Routers for eHealthCare Wireless Sensor Networks. In WiCOM'11, pages 1–4, 2011.

[9]M.Ameenet al. Smart e-Health Gateway: Bringing Intelligence to IoT-Based Ubiquitous Healthcare Systems. JMS, 36(1):93–101, 2012.

[10] K. Malasri et al. Addressing Security in Medical Sensor Networks. pages 7–12, 2007.

[11] X. Hung et al. An Efficient Mutual Authentication and Access Control Scheme for WSN in Healthcare. JN, 6(3):355–364, 2011.

Web reference:

[12]http://www.infosec.gov.hk/english/promotion/files/Script_common_auth
entication_methods_US.pdf, Accessed on 29-10-2014.

[13] Intel. Intel IoT Gateway, 2014. http://www.intel.com/content/www/us/en/embedded/products [accessed 2014-01-22].