An abstract security pattern for Zero Trust Access Control

ANDREI BRAZHUK, Yanka Kupala State University of Grodno (Belarus)
EDUARDO B. FERNANDEZ, Florida Atlantic University (USA)

Information systems have become very complex due to their increasing distribution, interconnection, and need to support complex communication structures. Zero Trust Architecture (ZTA) has been introduced as a solution to the security problems of complex networked systems. We present here a pattern to enforce the application of security controls on incoming requests from a variety of locations addressing data and services that are also heavily distributed. The Abstract Zero Trust Access Control pattern restricts access to the resources of a system by authenticating every network access request and enforcing authorization constraints to access specific resources; its enforcement applies two security principles: least privilege and complete mediation. While there are several other patterns that can be applied to implement a Zero Trust Architecture, this is its most fundamental pattern.

Categories and Subject Descriptors: D.2.11 **[Software Engineering]** Software Architectures–Patterns;

General Terms: Security Patterns, Secure Communications

Additional Key Words and Phrases: Abstract Security Pattern, Secure Channel, Authentication, Authorization, ZTA

Brazhuk, A. and Fernandez E.B. 2022. An abstract security pattern for Zero Trust Access Control. HILLSIDE Proc. of 29th Conf. on Pattern Lang. of Prog. (October 2022), 4 pages.

1. INTRODUCTION

Zero Trust Architecture (ZTA) has been introduced as a solution to the security problems of complex systems (Rose et al.,2020), based on the idea that no access requests for resources are trusted regardless of location, device from where they are sent, or role of the subject making the request. However, ZTA is a strategy for creating secure systems, it is not a ready-to-use architecture, although several implementations already exist. As a basic stage for the implementation of ZTA, we present a pattern that encapsulates the idea of complete network access control in order to prevent unauthorized access to the institution resources, which can be system resources such as operating system services, web resources, or application data. Several other patterns have been found in ZTA (Fernandez and Brazhuk 2022), most of which have already been published (Fernandez 2013). Our pattern is an example of an Abstract Security Pattern (ASP), a type of pattern that describes a conceptual security mechanism that realizes one or more security policies able to control (stop or mitigate) a threat or comply with a security regulation or policy (Fernandez et al., 2022); that is, we are not concerned with implementation aspects. We use the POSA template (Buschmann et al., 1996). Our audience includes security researchers, secure system designers, and security students.

2. ABSTRACT ZERO TRUST ACCESS CONTROL

2.1 Intent

The Abstract Zero Trust Access Control pattern restricts access to the resources of a system (system resources and data) by authenticating every network access request and enforcing authorization constraints to access resources. Its enforcement applies two security principles: least privilege and complete mediation.

2.2 Example

A company has moved some applications to the cloud. Previously hosted in an internal network, applications on a public space with broad anonymous access have been having security incidents. A recent attack was performed by a botnet, sending fake requests to a high-load application that led both to performance overhead and increasing

Author's address: Andrei Brazhuk; Yanka Kupala State University of Grodno (Belarus); email: andrew.brazhuk@gmail.com; Eduardo B.Fernandez; Florida Atlantic University (USA); email: fernande@fau.edu

charges for cloud use. The company is looking for a way of protecting its applications from unauthorized access from the internet and to enable secure employees' access to the applications, independently from the location of an application (internal network, cloud) or of the user (office, home).

2.3 Context

Institutions may own a complex and heterogeneous IT (Information Technology) infrastructure, using IoT (Internet of Things), cloud services, and cyber-physical systems; also, they may follow operational trends like BYOD (Bring Your Own Device), and WFH (Work From Home). This variety of networks has a large variety of distributed services and data and may receive requests from local and remote users, some registered and possibly many unknown. The heterogeneity of the devices used to access resources (phones, laptops, servers), the heavy distribution of subjects and resources, and the low security of home networks result in a large attack surface.

2.4 Problem

In modern institutions, users require access to a variety of resources and applications deployed across a distributed infrastructure. Most institutions use a perimeter-based security approach, which authenticates subjects and authorizes access to its resources by registered internal or specific external subjects; in many sites, once a subject is allowed in the system it has many unrestricted access rights. The new networks require access from registered subjects and by subjects who are not registered but have rights to access resources of the intranet (remote workers, cloud-based services). New systems need also to add finer access control to protect business assets from increasing and more sophisticated remote attacks.

The following forces affect a solution of this problem:

- *Heterogeneity*. Typical networks include devices from many origins, subjects from many locations, and resources from many types and locations.
- *Identity*. We need to keep track of the devices, subjects, and resources that are under our control or participating in our network. Knowing who is in the network improves security.
- *Compliance*. Some of the information in our system may have to follow regulations or industrial standards. These regulations may require access by only some types of users; for example, medical records may be accessed only by healthcare personnel.
- *Threats*. Attackers may make attempts to access resources, tamper with the contents of data, or introduce a malicious entity into a network. We need to stop these threats.
- *Overhead*. Threat control should not result in a large overhead.
- *Variability*. Resources may be frequently added or removed.
- *Variety of access control models*. Different systems use the Access Matrix, Role-Based Access Control, or other models (See Fernandez, 2013 for their patterns).
- *Least privilege*. We want to apply the principle of least privilege, where access to resources or data must be based on need-to-know policies where a subject is only given enough rights to perform its functions.
- *Complete mediation*. We want to apply the principle of complete mediation, where every attempt to access resources must be mediated and checked.

2.5 Solution

The network must intercept every access attempt for resources (complete mediation), authenticate them, and if they are legitimate (defined using a least privilege policy) grant access to them. Each access transaction must be logged.

2.5.1 *Structure.* We use the terminology and components that come from the NIST logical architecture of a ZTA (Rose et al., 2020). Fig. 1 shows the class diagram of the solution. An access Request from a Subject is received by the Policy Enforcement Point (PEP), which sends it to the Policy Decision Point (PDP). The PDP produces a Decision allowing or denying access, assisted by the Policy Engine (PE) which has the subject's verification information (identity, environment) and access to the authorization rules in the Policy Database. The authorization rules define rights following a least privilege policy. The decision of the PDP is sent to the PEP which then enables a Secure Communication Path (SCP) to let the subject apply an operation on the resource. The PDP also logs the access transactions.

2.5.2 *Dynamics.* Fig. 2 shows the use case “Validate access request”. It assumes that the subject has been previously authenticated in another use case, although it could be part of the same use case. Other use cases include "Monitor communication" and "Disable communication". The scenario starts with the authenticated subject requesting access to the PEP to perform the operation defined by “accessType” on a resource. The PEP forwards the request to the PDP that asks the PE if there is a rule indicating that the subject is authorized to access this resource in this way. With this information, the PDP decides that the subject is authorized and enables the PEP to configure the SCP to let the subject access the resource. The PDP also logs this transaction.

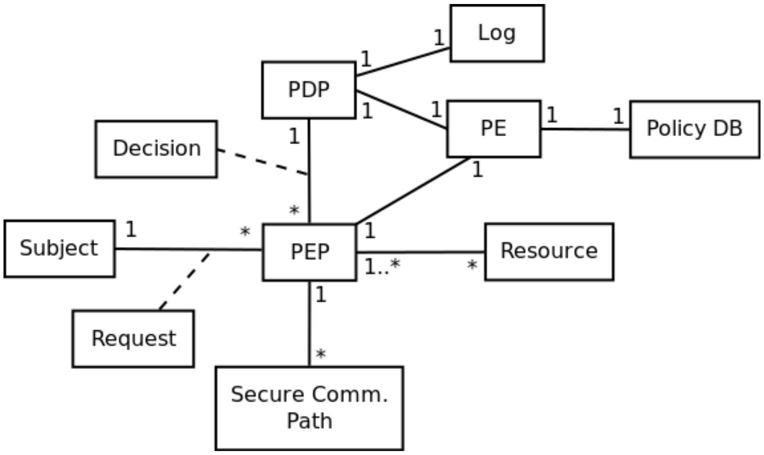

Fig. 1. Class diagram of the Abstract Zero Trust Access Control.

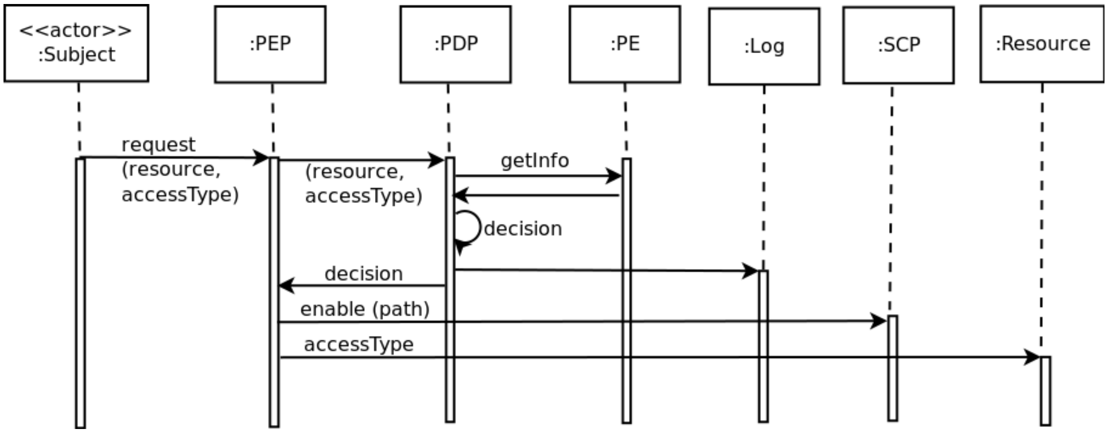

Fig. 2. Sequence diagram of use case “Validate access request”

2.6 Implementation.

Ref. (Rose et al.) shows several possible implementations which correspond to concrete patterns derived from our abstract pattern.

2.7 Known uses

Due to the fact that there are few implementations of this pattern, the solution has been partly defined by deduction; in this case, by the application of the two basic principles: complete mediation (the PEP intercepts all accesses) and least privilege (the rules in the policy database can be defined following a least privilege policy). A few implementations are now appearing, and the model may be refined later.

- The Black Core network for the military GIG (Global Information Grid) (Enterprise DoD 2007) implements a sort of least privilege model where a device or application is authenticated before access is granted access.
- Forrester's ZTA aims to move security mechanisms as close as possible to end network connections and implements micro-segmentation using a Security Gateway (SG) (Kindervag 2010). Network devices like UTM

(Unified Threat Management) and NGFW (Next Generation Firewall), sold by various vendors, can be considered SG implementations. Based on network-layer authentication (for example, IEEE 802.1X) a subject can be directed to the microsegment where the resource can be found.

- Google has implemented the BeyondCorp project (Ward et al., 2014), which objective was to remove the requirement for a privileged perimeter and enable access to applications from anywhere. They use a trust engine that evaluates the subject trust based on user and device (certificate-based) identity. This project defines authorizations based on the applications and not on network restrictions.
- SDP (Software Defined Perimeter) matches this pattern and uses SPA (Single Packet Authorization) as a network layer authentication mechanism and secures communications using mTLS (mutual TLS) (Kumar et al. 2019).

2.8 Consequences

This pattern provides the following benefits:

- *Heterogeneity*. The security mechanism may include devices, subjects, and resources from many origins and types.
- *Identity.* We can keep information about the devices, subjects, and resources under our control or that participate in our network.
- *Compliance.* We can keep track of restrictions on data based on compliance regulations.
- *Threats.* Authentication prevents intruders from getting into our system, authorizations restrict authenticated subjects to only specific resources.
- *Variability.* Changes in resources can be handled by changes in the policy databases.
- *Variety of access control models*. The policy database can contain rules according to any authorization model.
- *Least privilege*. Configuring the authorization rules appropriately can enforce least privilege policies.
- *Complete mediation*. By sending every access request to the PEP we can apply the principle of complete mediation.

The pattern has the following liabilities:

- It is important to select authorization models such as Role-Base Access Control or there may be a proliferation of rules.
- Authentication followed by authorization implies some overhead, but careful implementations can make it tolerable.

2.9 Example resolved

Every access request is now mediated and after authentication it is validated using this pattern. Every subject receives minimum rights according to its functions. Unauthorized access to resources has been stopped.

2.10 Related patterns

- Application Firewall (Fernandez, 2013). The application firewall filters calls and responses to/from enterprise applications, based on an institution's access control policies. It also has a content inspector that looks for malformed requests.
- XACML Access Control Evaluation pattern (Fernandez, 2013). XACML enables an organization to represent authorization rules in a standard manner, allowing a variety of rule combinations.
- Abstract Secure Communication Path (Fernandez and Brazhuk 2022b). The ASCP pattern describes how to construct a secure path between two endpoints, providing confidentiality, integrity, and authenticity. An endpoint is any device that is physically an endpoint on a network (laptop, phone, server) and which has an interface exposed to the system.
- Authorizer (Fernandez 2013). In an environment in which we have resources whose access needs to be controlled, describe who is authorized to access specific resources and in what way.
- Authenticator (Fernandez 2013). When a subject identifies itself to the system, how do we verify that the subject intending to access the system is the one it says it is?

ACKNOWLEDGMENTS

We thank our shepherd, Prof. Federico Balaguer, for his insightful comments that significantly improved this paper. The participants in the Writers' Workshop at PLoP 2022 (Francisco Gindre, André Cordeiro, Matías Urbieta, Valentino Vranic, Pavel Hruby, and Mary Tedeschi) provided valuable ideas to improve this paper.

REFERENCES

F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal. Pattern-Oriented Software Architecture: A System of Patterns, Vol. 1. J. Wiley, 1996.

DoD Enterprise, "Department of Defense Global Information Grid Architectural Vision." 2007.

E.B.Fernandez, Security Patterns in Practice: Designing Secure Architectures Using Software Patterns. Wiley, 2013.

E.B.Fernandez, N.Yoshioka, H. Washizaki, J. Yoder, "Abstract security patterns and the design of secure systems", Cybersecurity, April 2022. https://doi.org/10.1186/s42400-022-00109-w

E.B.Fernandez, A. Brazhuk, "A critical analysis of Zero Trust Architecture (ZTA)". Preprint, 2022

E.B.Fernandez, A.Brazhuk, "The Abstract Secure Communication Path (ASCP) pattern and a derived VPN pattern". Procs. of the Latin American PLoP 2022.

P.Kumar, A.Moubayed, A.Refaey, A. Shami, J.Koilpillai, "Performance analysis of SDP for secure internal enterprises". 2019 IEEE Wireless Communications and Networking Conference (WCNC), pp. 1-6. IEEE 2019.

S.Rose, O.Borchert, S.Mitchell, S. Connelly, "Zero Trust Architecture", Special Publication (NIST SP-800-207), August,2020. https://doi.org/10.6028/NIST.SP.800-207

J.H. Saltzer, M.D. Schroeder, "The protection of information in computer systems". Proceedings of the IEEE, 1975, 63(9).

R. Ward, B. Beyer, "BeyondCorp: A New Approach to Enterprise Security". Login 2014, vol. Vol. 39, No. 6, pp. 6-11