Bezpieczeństwo w inżynierii oprogramowania

Architektura zerowego zaufania

Arkadiusz Chrobot

Katedra Systemów Informatycznych

23 listopada 2024

Plan

Wstęp

Zasada *zerowego zaufania* (ang. *zero trust*) i wynikająca z niej *architektura zerowego zaufania* (ang. *zero trust architecture*) nie są w informatyce nowymi koncepcjami, ale w ostatnich latach zyskują na znaczeniu. Związane jest to z istotnymi zmianami w sposobie funkcjonowania i organizacji systemów informatycznych. Tradycyjnie ochrona takich systemów dotyczy ich obrzeża (ang. *perimeter*) [1]. Jeśli intruz pokona zabezpieczenia perymetru, to zyskuje możliwość prawie nieograniczonej eksploracji jego wnętrza. Zatem nie jest to dobre podejście. Dodatkowo kwestię bezpieczeństwa komplikuje wprowadzenie usług opartych o technologie chmurowe, mikroserwisów i urządzeń mobilnych (zasada *Bring Your Own Device — BYOD*). Ponadto statystyki ataków cybernetycznych wskazują, że najczęściej ich źródłem są intruzi wewnętrzni (ang. *insiders*). Te czynniki zadecydowały o konieczności zmiany sposobu projektowania systemów informatycznych.

Definicje

Zasada zerowego zaufania

Zasada *zerowego zaufania* (ang. *zero trust*) to zbiór koncepcji zaprojektowanych by zminimalizować niepewność w egzekwowaniu precyzyjnych decyzji, o udzielaniu dla każdego żądania dostępu o minimalnym stopniu uprzywilejowania, do usług i systemów informatycznych, przy założeniu, że zabezpieczenia sieci komputerowej zostały przełamane.

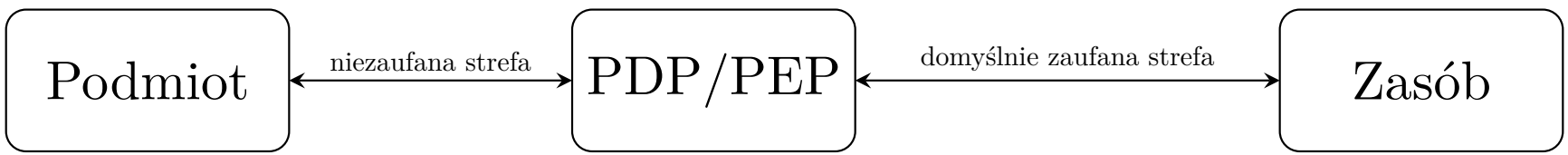
Architektura zerowego zaufania

Architektura zerowego zaufania (ang. *zero trust architecture*) jest projektem cyberbezpieczeństwa, który wykorzystuje koncepcje zerowego zaufania i obejmuje relacje między komponentami, planowanie przepływu prac i polityki bezpieczeństwa.

Architektura zerowego zaufania dotyczy całej struktury systemu informatycznego, której istotnym elementem jest oprogramowanie.

Definicje

Rysunek na slajdzie przedstawia abstrakcyjny model dostępu do zasobów w architekturze zerowego zaufania. *Podmiot* (ang. *subject*) to użytkownik lub urządzenie, które chce uzyskać dostęp do zasobu. Komponent PDP/PEP to *punkt decyzyjny polityki bezpieczeństwa* (ang. *Policy Decision Point*) i odpowiadający mu *punkt egzekwowania polityki* (ang. *Policy Enforcement Point*).



Każde żądania podmiotu w stosunku do zasobu musi przejść przez komponent sprawdzania/egzekwowania polityki bezpieczeństwa, czyli za każdym razem podmiot podlega uwierzytelnieniu i autoryzacji. Celem tego mechanizmu jest *zminimalizowanie ryzyka*, że dostęp do zasobu zostanie udzielony nieuprawnionemu podmiotowi.

Zasady

Architektura zerowego zaufania jest projektowana i wdrażana zgodnie z następującymi podstawowymi zasadami:

1. Wszystkie dane i usługi są uważane za zasoby.
2. Cała komunikacja jest zabezpieczona, niezależnie od lokalizacji sieci.
3. Dostęp do poszczególnych zasobów przedsiębiorstwa jest przyznawany w ramach sesji (ang. *per-session*).
4. Dostęp do zasobów jest określany na bazie dynamicznie zmienianej polityki, opartej na postrzeganym stanie tożsamości klienta, aplikacji/usługi i wnoszącym żądanie aktywom (ang. *asset*) i mogącej uwzględniać atrybuty środowiska i zachowania.
5. Przedsiębiorstwo monitoruje i bada integralność i poziom bezpieczeństwa wszystkich posiadanych i stowarzyszonych aktywów.

Zasady

6. Uwierzytelnienie i autoryzacja dostępu do zasobów jest przeprowadzana w sposób dynamiczny i ściśle przestrzegany.
7. Przedsiębiorstwo gromadzi tyle informacji, ile jest to możliwe o bieżącym stanie aktywów, infrastrukturze sieciowej oraz komunikacji i używa jej by poprawić stan bezpieczeństwa (ang. *security posture*).

Dodatkowo przyjmowane są następujące założenia odnośnie do infrastruktury sieciowej:

1. Cała prywatna sieć przedsiębiorstwa nie jest domyślnie uważana za zaufaną strefę.
2. Urządzenia w sieci mogą nie należeć do przedsiębiorstwa lub ich konfiguracja może nie być przez przedsiębiorstwo zarządzania.

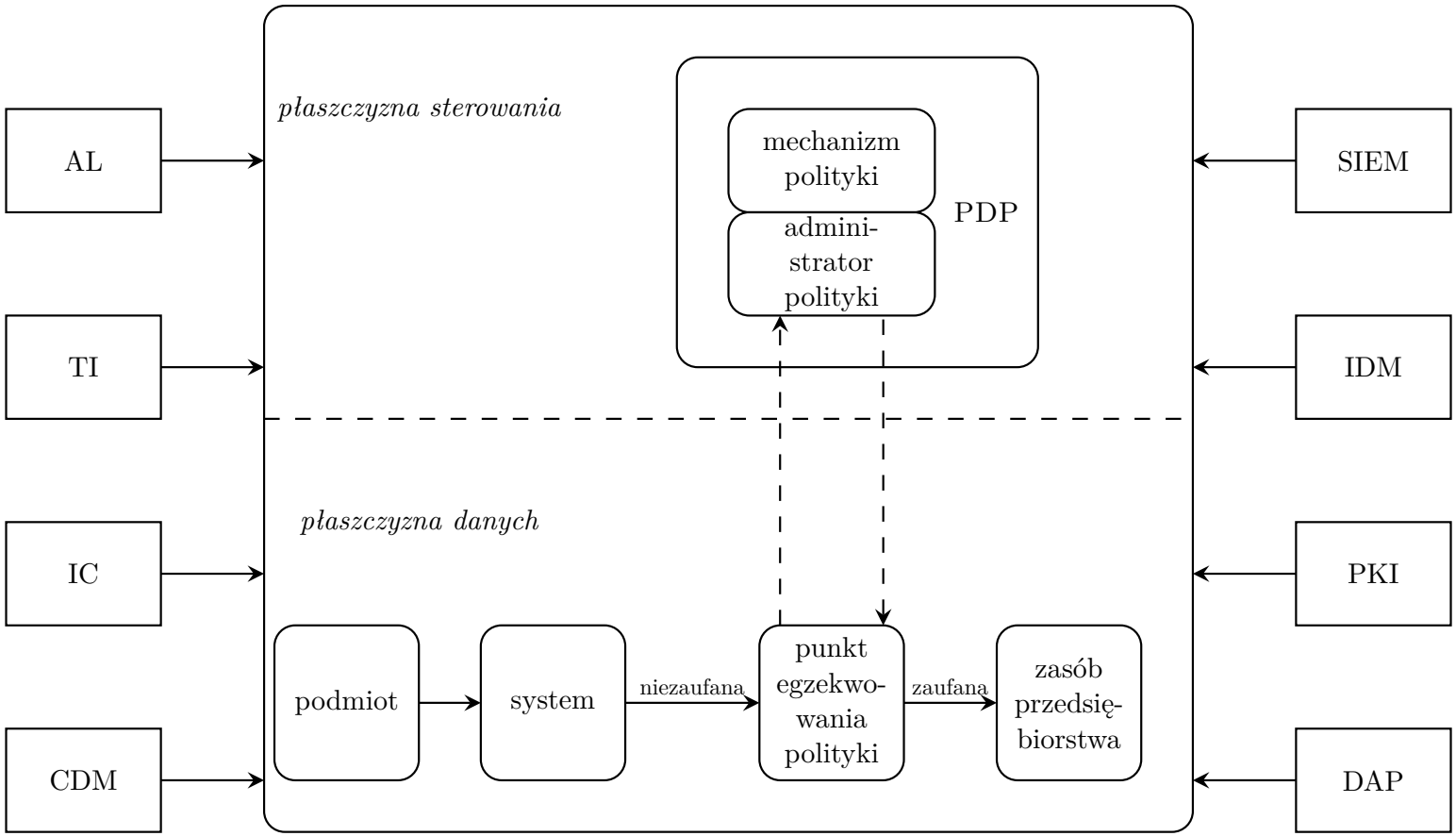
Zasady

3. Żaden zasób nie jest domyślnie zaufanym.
4. Nie wszystkie zasoby przedsiębiorstwa są w infrastrukturze należącej do niego.
5. Zdalne podmioty i aktywa przedsiębiorstwa nie mogą w pełni ufać lokalnej sieci, w której się znajdują.
6. Aktywa i zadania (ang. *workflow*) przemieszczające się między infrastrukturą należącą i nienależącą do przedsiębiorstwa mają spójną politykę i poziom zabezpieczeń.

Komponenty

Rysunek 1 przedstawia podstawowe zależności między komponentami architektury zerowego zaufania oraz ich interakcję. Punkt decyzyjny polityki (PDP) został podzielony na dwa logiczne komponenty: *mechanizm polityki* (ang. *policy engine*), odpowiedzialny za podejmowanie ostatecznej decyzji o przyznaniu dostępu do zasobu dla danego podmiotu, oraz *administratora polityki* (ang. *policy administrator*), który ustanawia lub przerywa ścieżkę komunikacji między podmiotem i zasobem. Oba komponenty ze sobą ściśle współpracują. Jeśli mechanizm polityki zadecyduje o przyznaniu dostępu, to administrator wygeneruje odpowiedni token sesyjny i skonfiguruje punkt egzekwowania polityki (PEP), aby pozwolił sesji komunikacji się rozpocząć. Punkt egzekwowania polityki jest odpowiedzialny za nawiązanie, monitorowanie i ostatecznie zamknięcie połączenia między podmiotem, a zasobem przedsiębiorstwa.

Komponenty



Relacje komponentów architektury zerowego zaufania

Komponenty

Komponenty architektury zerowego zaufania używają osobnej *płaszczyzny sterowania* do komunikacji, a dane aplikacji są wymieniane w obrębie *płaszczyzny danych*. Dodatkowo, te komponenty mogą używać zewnętrznych źródeł danych do podejmowania decyzji. Do tych źródeł zalicza się:

CDM system ciągłej diagnostyki i zapobiegania (ang. *continuous diagnostics and mitigation*), który zbiera informacje o aktywach przedsiębiorstwa i wprowadza aktualizacje konfiguracji i oprogramowania,

IC system zgodności ze standardami przemysłowymi (ang. *industry compliance*), który zapewnia zgodność przedsiębiorstwa z regulacjami dotyczącymi określonej branży,

TI system analizy zagrożenia (ang. *threat intelligence*), który dostarcza danych o najnowszych zagrożeniach, atakach i podatnościach.

Komponenty

AL dzienniki aktywności sieciowej i systemowej,

DAP polityki dostępu do danych (ang. *data access policies*),

PKI struktura klucza publicznego przedsiębiorstwa,

IDM system zarządzania tożsamościami/identyfikatorami (ang. *ID management*),

SIEM system rejestrowania informacji o zdarzeniach związanych z bezpieczeństwem.

Warianty architektury

Istnieje kilka wariantów architektury zerowego zaufania, które różnią się komponentami używanymi jako główne źródło reguł polityki bezpieczeństwa. W niektórych z nich nacisk kładziony jest na komponenty zapewniające *ulepszone zarządzanie tożsamością* (ang. *enhanced identity governance*), w których polityki bazują na tożsamości i atrybutach przypisanych do podmiotów. Dostęp do zasobów jest udzielany podmiotom przez PEP po ich uwierzytelnieniu i przeprowadzeniu autoryzacji. Wadą tego rozwiązania jest to, że potencjalni intruzji mogą uzyskać wstępny dostęp do sieci i wykonać rekonesans lub zapoczątkować atak typu DoS.
Inne rozwiązania bazują na mikro-segmentacji sieci. W tych przypadkach sieci jest dzielona na małe segmenty z zasobami, które są chronione przez sieciowe urządzenia dostępowe (ang. *gateway*). Rolę takich urządzeń mogą spełniać też programowi agenci (ang. *software agents*).

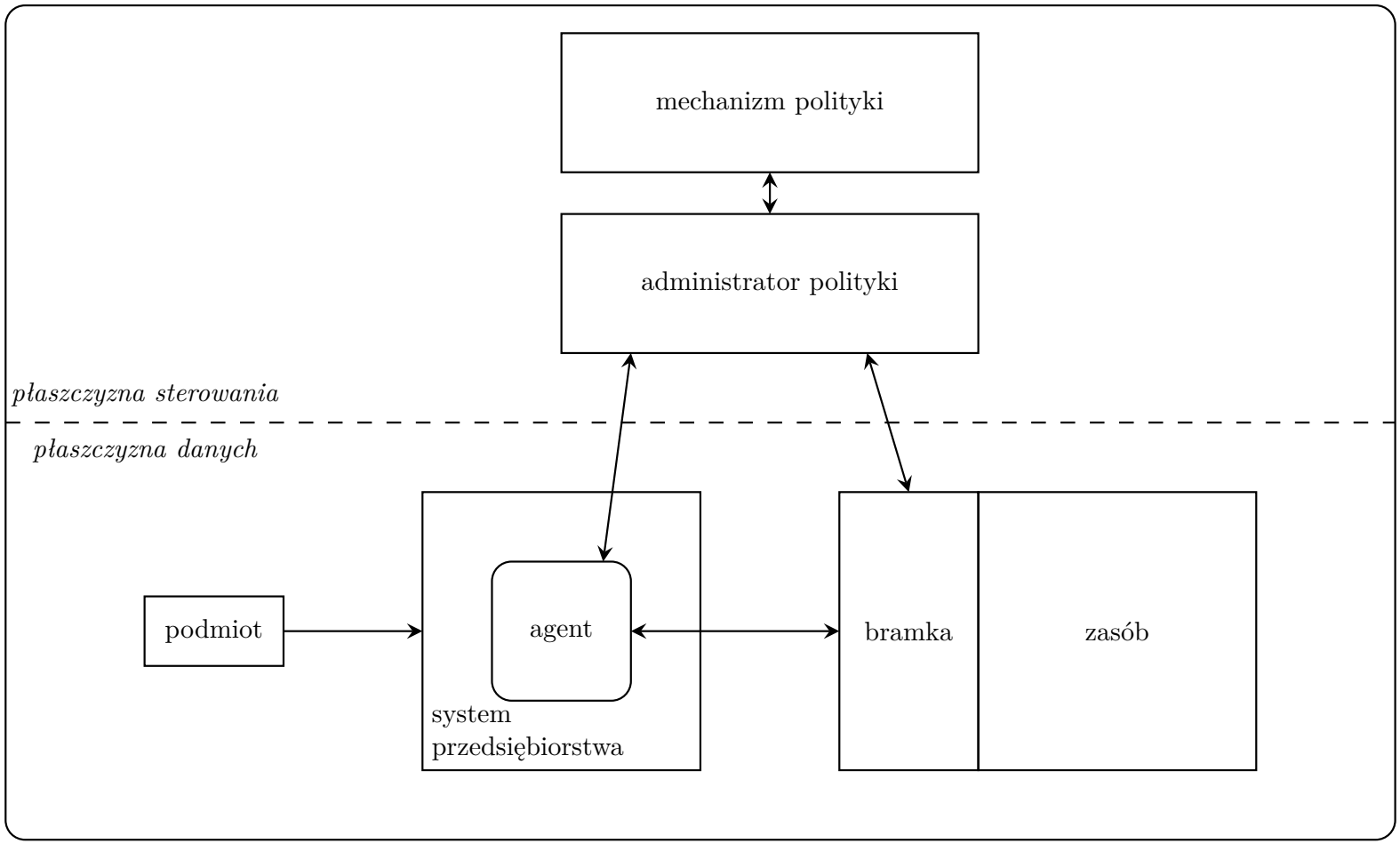
Warianty architektury

Istotą tego podejścia jest konieczność użycia komponentów PEP (wspomnianych urządzeń lub agentów), które są zarządzane i powinny reagować i rekonfigurować się w odpowiedzi na zagrożenia lub zmianę poziomu obciążenia.
Trzeci wariant architektury zerowego zaufania jako główny komponent wykorzystuje infrastrukturę sieciową i perymetr definiowany programowo (ang. *software defined perimeter*). Rozwiązanie to bazuje na sieciach nakładkowych (ang. *overlay networks*). Administrator polityki działa jako kontroler sieci, który ją zestawia i konfiguruje bazując na decyzjach podejmowanych przez mechanizm polityki.

Przykłady architektur

W modelu agent urządzenia/bramka (ang. *device agent/gateway*) punkt egzekwowania polityki (PEP) jest podzielony dwa komponenty, które ulokowane są na zasobie lub przed chronionym zasobem (Rysunek 2). Na wszystkich aktywach jest umieszczony agent koordynujący połączenia, a przed zasobem jest bramka pełniąca rolę pośrednika (ang. *proxy*). Agent jest komponentem programowym, który kieruje ruch do odpowiedniego PDP, aby żądania aktywów zostały sprawdzone. Z kolei bramka komunikuje się z administratorem polityki, aby dopuścić do zasobu tylko te ścieżki komunikacji, które zostały skonfigurowane i zatwierdzone przez administratora.

Przykłady architektur

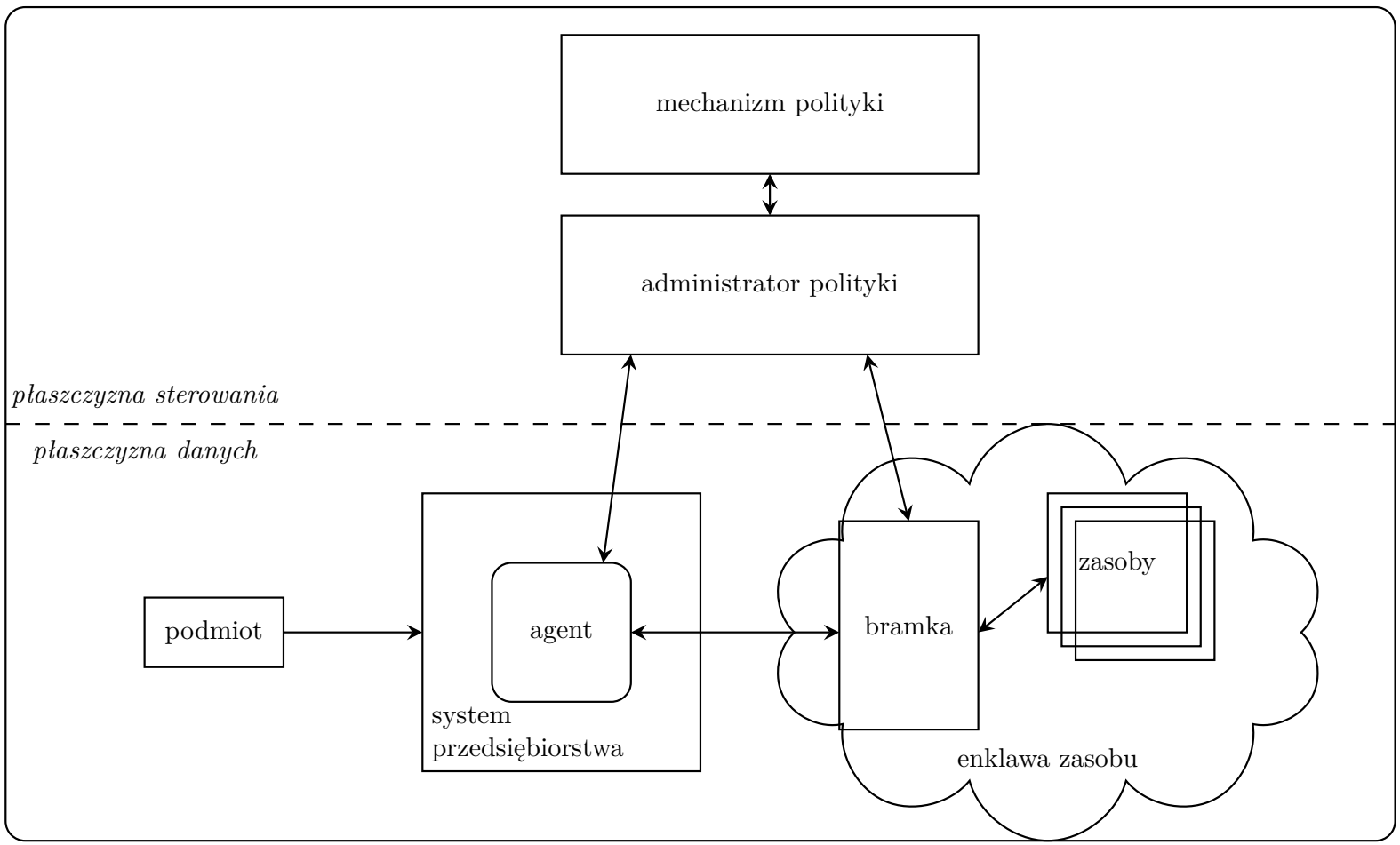


Model typu agent urządzenia/bramka (ang. *device agent/gateway*)

Przykłady architektur

Model bramki enklawy (ang. *enclave gateway*) jest wariantem modelu agent urządzenia/bramka (ang. *device agent/gateway*). W tym modelu bramka ulokowana jest na granicy enklawy zasobów (Rysunek 3). Zazwyczaj zasoby w enklawie służą jednej funkcji biznesowej lub nie mogę się bezpośrednio komunikować z bramką ze względu na swoje ograniczenia (np. może to być przestarzałe oprogramowanie, które jednak ciągle jest używane produkcyjnie). Ten model może również być użyteczny dla przedsiębiorstw, które bazujących na rozwiązaniach chmurowych mikrousług, w celu realizacji poszczególnych procesów biznesowych.

Przykłady architektur

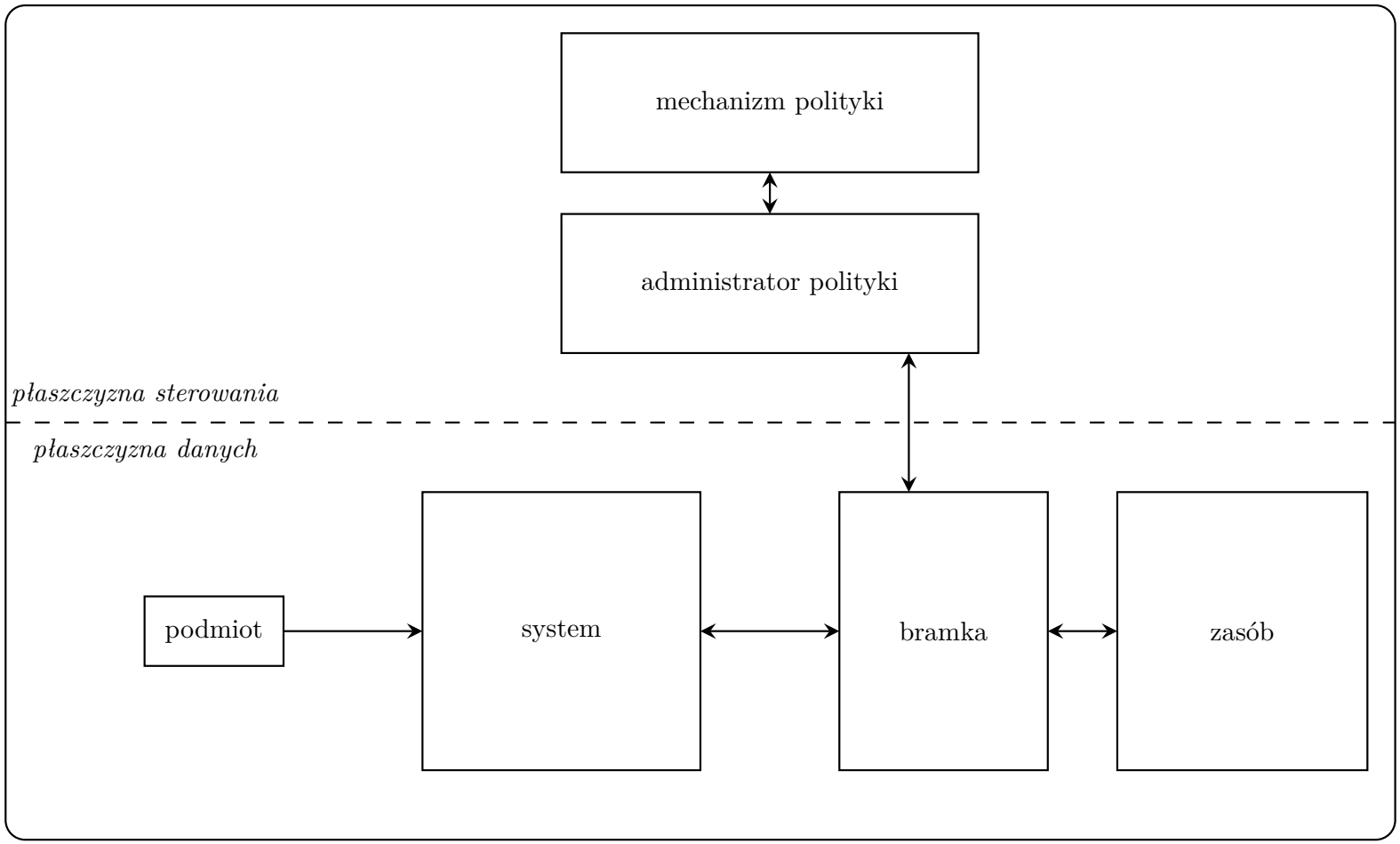


Model typu brama enklawy (ang. *enclave gateway*)

Przykłady architektur

Model portalu zasobu (ang. *resource portal model*) punkt egzekwowania polityki jest pojedynczym komponentem, który pełni rolę bramki dla wszystkich żądań podmiotu (Rysunek 4). Przykładem może być portal do prywatnego środowiska chmurowego lub centrum obliczeniowego zawierającego odziedziczone (ang. *legacy*) aplikacje. Zaletą tego rozwiązania jest to, że nie trzeba na wszystkich aktywach instalować dodatkowego oprogramowania. Jest ono także odpowiednie dla polityk dotyczących urządzeń mobilnych i międzyorganizacyjnych projektów. Wadą jest to, że można uzyskać tylko ograniczoną informację na temat zewnętrznych urządzeń przyłączanych do systemu. Ten model jest także narażony na ataki typu DoS, więc musi posiadać odpowiednio duże zasoby, aby nim przeciwdziałać.

Przykłady architektur



Model typu portal zasobu (ang. *portal zasobu*)

Algorytm zaufania

Implementacja architektury zerowego zaufania wymaga użycia algorytmu zaufania (ang. *trust algorithm*), który jest wykonywany przez mechanizm polityki, w celu podjęcia decyzji o przyznaniu lub odmowie dostępu do zasobu. Dane wejściowe dla takiego algorytmu mogą pochodzić z wielu źródeł (analiza zagrożeń, dane historyczne o zachowaniu podmiotu, itd.). Istnieje kilka wariantów takich algorytmów. Niektóre biorą pod uwagę zdefiniowane przez przedsiębiorstwo kryteria, które są traktowane jako warunki do spełnienia. Inne dla podmiotów wyliczają ocenę (ang. *score*) zaufania i na jej podstawie podejmują decyzje. Część z nich weryfikuje każde żądanie z osobna, a pozostałe dokonują tej oceny w oparciu o historyczny kontekst, tj. biorą pod uwagę poprzednie żądania danego podmiotu.

Zagrożenia

Architektury zerowego zaufania nie są pozbawione zagrożeń. Oto niektóre z nich:

- sabotaż procesu decyzyjnego (administratora polityki i mechanizmu polityki),
- ataki typu DoS lub zakłócanie działania sieci,
- kradzież poświadczeń/wewnętrzne zagrożenia,
- widoczność ruchu sieciowego,
- składowanie danych o systemie i sieci,
- zależność od własnościowych formatów danych lub rozwiązań,
- użycie automatycznych narzędzi do administracji implementacją architektury zerowego zaufania.

Źródła I

Scott Rose, Olivier Borchert, Stu Michell i Sean Connelly. Zero Trust Architecture. URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf (termin wizyty 12.11.2024).

Pytania

?

Koniec

Dziękuję Państwu za uwagę!