Przegląd technologii wspomagających Internet rzeczy

Wstęp

Internet rzeczy (IoT) początkowo wykorzystywał obecną infrastrukturę internetową i istniejące technologie do przekształcania samodzielnych obiektów (tj. urządzeń) w połączone ze sobą inteligentne obiekty. Przed opracowaniem technologii specyficznych dla IoT, komputery i technologie sieciowe były stosowane, z poważnymi poprawkami, do obsługi aplikacji IoT. Na przykład protokół IPv4, który był używany w bezprzewodowych sieciach czujników (WSN), był szeroko stosowany na początku IoT do łączenia węzłów z Internetem za pomocą jednego adresu IP (tj. Translacji adresów sieciowych (NAT)), co wymaga dodatkowego wysiłku przy konfiguracji. Jednak wiele problemów wynikło z korzystania z technologii, które nie zostały zaprojektowane z myślą o IoT (np. brak autokonfiguracji w IPv4). W związku z tym istniała potrzeba opracowania nowych technologii w celu rozwiązania różnych problemów, takich jak wydajny routing, skalowalność i mobilność, aby tworzenie aplikacji IoT było łatwiejsze i bardziej wydajne. Te postępy w technologiach IoT zaowocowały szeroko zakrojonym wdrażaniem aplikacji IoT w świecie rzeczywistym w wielu dziedzinach, takich jak opieka zdrowotna, przemysł i inteligentne budynki. Biorąc pod uwagę rozwój technologii IoT w ciągu ostatniej dekady, jasne jest, że najbardziej wyraźne trendy dotyczą wydajności, która wymaga optymalnego wykorzystania zasobów (np. baterii i pamięci). Niskie zużycie energii jest ważnym czynnikiem przy projektowaniu nowych aplikacji IoT. Urządzenia oszczędzają energię, gdy są w trybie „uśpienia", i zużywają energię, gdy są aktywne (lub przebudzone). Urządzenia są zaprojektowane tak, aby wybudzały się zgodnie z optymalnym harmonogramem w celu wykonania niektórych operacji (tj. Zbieranie i wysyłanie danych). Złożoność tych operacji musi zostać zoptymalizowana, aby zwiększyć żywotność urządzeń. Projektanci dążą do osiągnięcia niskiej złożoności urządzenia, niskiego zużycia energii (tj. długiej żywotności sieci) oraz odpowiedniej równowagi między możliwościami przetwarzania sygnału/danych a komunikacją. Ważną kwestią dotyczącą wydajności jest brak interoperacyjności w IoT (tj. urządzenia różnych marek i modeli nie mogą dobrze współpracować bez warstwy tłumaczeniowej). Kwestia interoperacyjności pozostaje głównym wyzwaniem w obecnych technologiach IoT ze względu na brak ujednoliconych standardów dla IoT. Mamy nadzieję, że wraz z dojrzewaniem IoT problemy te zostaną zminimalizowane. W tym rozdziale szczegółowo opisano kluczowe koncepcje technologii wspomagających na podstawie ich pozycji w dobrze znanym modelu pięciowarstwowym, który składa się z warstwy percepcji, warstwy sieci, warstwy oprogramowania pośredniego, warstwy aplikacji i warstwy biznesowej.

Przegląd architektury IoT

Zarządzanie milionami heterogenicznych urządzeń podłączonych przez Internet wymaga elastycznej, warstwowej architektury. Jak przedstawiono w Części 1, istnieją różne modele architektoniczne IoT, a mianowicie Internet of Things - Architecture (IoT-A), Industrial Internet Reference Architecture (IIRA), Reference Architecture Model Industrie 4.0 (RAMI 4.0) oraz Cisco′s Internet of Things Reference Model. Nie przyjęto jednak żadnej standardowej architektury referencyjnej. Trwają również projekty mające na celu projektowanie nowych architektur dla IoT, takie jak P2413 - Standard for the Architectural Framework for the IoT1, European Research Cluster on the Internet of Things (IREC)2, Internet of Things Reference Architecture (IoT RA)3, i Ramy Arrowhead. Ponadto istnieją inne architektury, które są związane z IoT, ale skupiają się tylko na podzbiorach IoT. Na przykład Europejskie Normy Telekomunikacyjne Komitet Techniczny Instytutu (ETSI TC) ds. M2M koncentruje się wyłącznie na standardach komunikacyjnych, podczas gdy Sektor Normalizacji Telekomunikacyjnej Międzynarodowego Związku Telekomunikacyjnego (ITU-T)6 skupia się wyłącznie na systemach identyfikacji. W celu systematycznego omówienia technologii wspierających IoT, zostaną one omówione w oparciu o ich wygląd na modelu architektury. Na nasze potrzeby (tj. aby przedstawić

przegląd ogólnych technologii IoT), żadna z powyższych architektur referencyjnych nie jest odpowiednia z następujących powodów. Po pierwsze, IoT-A jest stosunkowo stary i nie definiuje kluczowych pojęć IoT, takich jak przetwarzanie w chmurze i mgła, które pozostały do zidentyfikowania podczas implementacji. Po drugie, IoT-A skupia się tylko na WSN i działaniach aplikacji i nie obejmuje innych technologii, które są ważne w nowoczesnych aplikacjach IoT (takich jak przetwarzanie Big Data do celów analitycznych). Po trzecie, RAMI 4.0 i IIRA są przeznaczone głównie dla Przemysłowego IoT (IIoT) i nie rozszerzają się na inne domeny IoT, takie jak Internet of Medical Things (IoMT), Internet of Vehicles (IoV) i tak dalej. Co więcej, model referencyjny Cisco nie może być wykorzystany, ponieważ nie został jeszcze sfinalizowany. Wreszcie, ponieważ technologie będą prezentowane bez konkretnego scenariusza IoT, odpowiedni jest pseudofizyczny punkt widzenia; jednak żadna z powyższych architektur nie zapewnia tego punktu widzenia. Dlatego w tej części dokonano kategoryzacji technologii wspierających IoT w oparciu o kompleksową, ogólną i prostą architekturę, a mianowicie pięciowarstwowy model IoT. Model ten ujmuje wiele esencji innych modeli (oraz upraszcza je i systematyzuje). Dlatego ta architektura była szeroko wykorzystywana w ostatnich publikacjach. Każda warstwa zawiera przykłady technologii, które są podzielone na kategorie według ich funkcjonalności. Warstwy te są koncepcyjnie podobne do warstw kontroli transmisji (TCP)/protokołu internetowego (IP), co zostanie szczegółowo omówione w następnej sekcji. Chociaż wiele technologii IoT może pasować do modelu TCP/IP (lub Open Systems Interconnection (OSI)), kategoryzacja wszystkich technologii IoT w oparciu o model TCP/IP nie obejmuje wszystkich technologii. W związku z tym bardziej odpowiednia jest ich klasyfikacja na podstawie modelu pięciowarstwowego.

Technologie wspomagające

Technologie wspierające IoT różnią się w zależności od domeny i scenariusza. Na przykład inteligentny transport wymaga elastycznych technologii, które zapewniają łączność ogromnej liczby węzłów mobilnych. Natomiast w opiece zdrowotnej koncentruje się na niezawodności i uczciwości. Dlatego tutaj pokrótce przedstawiono ogólne technologie w oparciu o ich funkcjonalność w pięciowarstwowym modelu IoT.

Technologie warstwy percepcji

Warstwa percepcji (znana również jako warstwa obiektów) jest pierwszą warstwą w modelu IoT. Obejmuje różnego rodzaju urządzenia fizyczne, które są odpowiedzialne za zbieranie danych i odpowiednie działanie, takie jak identyfikatory obiektów, pomiary temperatury, lokalizacji i wilgotności. Zużycie energii i zdolność komunikacyjna (tj. jednokierunkowa lub dwukierunkowa) to ważne aspekty tej warstwy. Klasyfikacja technologii warstwy percepcyjnej jest następująca:

Pasywna

Urządzenia pasywne mają najniższy zasięg radiowy, ponieważ nie mają wbudowanych zasilaczy i polegają na czytnikach znajdujących się w pobliżu w celu zasilania energią. Ograniczeniem pasywnych technologii wspomagających jest to, że komunikacja z nimi jest jednokierunkowa. Dane mogą być odczytywane z węzłów, ale nie mogą być zapisywane z powrotem do węzłów. Jednak w przypadku wielu aplikacji wystarcza komunikacja jednokierunkowa. Przykładami technologii pasywnej warstwy percepcyjnej są:

* Kody szybkiej odpowiedzi (QR). Te kody działają jako metoda przechowywania informacji w węźle IoT i mogą być używane do rozpoznawania węzłów. Kody QR to prawdopodobnie najłatwiejsza i najtańsza technologia w tej kategorii. Potencjalnie można je zastosować do prawie wszystkiego. Drukując kod o powierzchni 1 cala na węzłach, stają się one łatwo rozpoznawalne przy użyciu różnych czytników, takich jak kamery lub czujniki światła, chociaż muszą znajdować się w zasięgu wzroku

czytnika. Kody QR mogą być wykorzystywane w wielu scenariuszach, takich jak przesyłki lub śledzenie produkcji.

* Tagi pasywnej identyfikacji radiowej (RFID). Te znaczniki to małe elektroniczne chipy, które - podobnie jak kody QR - można przymocować do dowolnego węzła, umożliwiając rozpoznanie węzłów w celu śledzenia lub automatycznej identyfikacji, co umożliwia takie aplikacje, jak zarządzanie zapasami. Czytelnicy rozpoznają te chipy, gdy znajdują się w niewielkiej odległości, wysyłając sygnał, który zapewnia wystarczającą moc, aby tag zareagował; odpowiedź zawiera dane identyfikacyjne.

Półpasywny

Urządzenia półpasywne posiadają baterie, które zasilają tagi podczas odbierania sygnałów z czytnika. W przeciwieństwie do technologii pasywnych, dodatkowa dostępna moc umożliwia im odczytywanie i zapisywanie danych. Przykładami technologii semipasywnych są:

* Półpasywne tagi RFID. Tagi te są podobne do tagów pasywnych, ale ich baterie umożliwiają im większy zasięg, co umożliwia wiele dodatkowych zastosowań. Jednym z przykładów jest inteligentny transport, na przykład blokowanie przejazdów kolejowych, gdy zbliża się pociąg.

* Czytniki podczerwieni (IR). Czytniki te to urządzenia, które wyczuwają promieniowanie elektromagnetyczne niewidzialne dla ludzi. Nowoczesne zastosowania tej technologii obejmują automatyczne i zdalne wykrywanie temperatury ciała oraz zdalne sterowanie. W kontekście IoT podłączone czytniki podczerwieni mogą wyzwalać pewne działania (np. alarm) w przypadku wykrycia nieprawidłowości (np. wysokiej temperatury ciała). Ponieważ IR jest tanie, proste i zużywa mało energii, może być preferowane w niektórych scenariuszach IoT, w których czytniki IR mogą zostać skradzione (np. monitorowanie temperatury ciała na lotniskach). Jest jednak ograniczony do linii wzroku i podlega zakłóceniom lub uszkodzeniom z powodu obiektów blokujących sygnały świetlne.

Aktywny

Urządzenia aktywne często mają największy zasięg, ponieważ mają większe baterie lub są podłączone do zasilaczy, które umożliwiają im odbieranie i przesyłanie danych. Przykłady tych urządzeń są następujące:

* Aktywne tagi RFID. Podobny do półpasywnych tagów RFID, ale ma większy zasięg dzięki większej wbudowanej baterii. Znaczniki te są najczęściej używane w systemach lokalizacji w czasie rzeczywistym (RTLS)7, na przykład do monitorowania cennych zasobów (np. do śledzenia zgubionych lub skradzionych drogich urządzeń medycznych). W takich scenariuszach IoT połączone węzły wyposażone w aktywny RFID mogą wywołać alarm, gdy opuszczą określony obszar.

* Inteligentne siłowniki. Urządzenia mechaniczne, które mogą być sterowane automatycznie, bez interwencji człowieka, w celu zastosowania określonego ruchu do produktu lub procesu w oparciu o wykryte dane lub w oparciu o zdefiniowany przepływ pracy. Połączone siłowniki mogą pomóc przemysłowi lub inteligentnym miastom zwiększyć wydajność i zaoszczędzić pieniądze, ponieważ siłowniki mogą reagować automatycznie na podstawie wykrytych danych, co wymaga mniejszej liczby pracowników.

* Inteligentne czujniki do noszenia, wbudowane lub samodzielne. Urządzenia, które zawierają wbudowane czujniki, takie jak żyroskopy, radia z globalnym systemem pozycjonowania (GPS) i monitory tętna używane w nowoczesnych smartfonach lub inteligentnych zegarkach do monitorowania aktywności i stanu zdrowia użytkowników. Inteligentne termostaty wykorzystują również podłączone czujniki do efektywnego zarządzania energią. W nowoczesnych aplikacjach IoT

takie połączone urządzenia (np. smartwatch, smartfon itp.) często współpracują ze sobą, aby osiągnąć wspólny cel (np. wszechobecne monitorowanie zdrowia, ostrzeganie opiekunów itp.).

Technologie warstwy sieciowej

Drugą warstwą jest warstwa abstrakcji sieci lub obiektu. Jest uważany za warstwę infrastruktury, ponieważ jej technologie przekształcają konwencjonalne czujniki opisane w warstwie percepcji w inteligentne i połączone węzły. Technologie warstwy sieci umożliwiają identyfikację węzłów przez Internet lub dowolną sieć lokalną, co pozwala im na bezpieczną komunikację między sobą. Wiele technologii w tej warstwie znajduje się również w pierwszych trzech warstwach pakietu IP (TCP/IP) (tj. Link, Internet i Transportlayers). Ze względu na ograniczone możliwości (tj. moc i moc obliczeniową) większości węzłów IoT, w celu zapewnienia interoperacyjności między urządzeniami IoT wymagane są skalowalne i wydajne techniki routingu. Należy zauważyć, że wiele technologii IoT jest faktycznie wykorzystywanych w sieciach WSN lub komunikacji maszyna-maszyna (M2M), ale zostały one ulepszone w celu spełnienia wymagań IoT (tj. łączności z Internetem). Wiele nowoczesnych urządzeń wykorzystuje więcej niż jedną technologię; jak w inteligentnych zegarkach, które często mają Wi-Fi, Bluetooth i NFC. W tej sekcji kategoryzujemy technologie wspomagające na podstawie ich funkcjonalności w warstwie sieciowej.

Identyfikacja

Gdy węzeł łączy się z siecią, jest mu przypisywany identyfikator, który umożliwia mu komunikację z innymi węzłami. Identyfikacja jest ważna, aby kontrolować nadmierne wykorzystanie przepustowości w dużych sieciach (np. zalanie) poprzez zapewnienie, że komunikują się tylko zidentyfikowane węzły. Aby łatwo zlokalizować urządzenia w ogromnej sieci, do urządzeń przypisywane są nazwy i adresy. Konwencje nazewnictwa przypisują strukturalne nazwy do węzłów; nazwy te umożliwiają łatwą identyfikację węzłów i mogą nawet szczegółowo opisywać funkcje. Rozwiązania nazewnictwa są następujące:

* Jednolity identyfikator zasobu (URI)8. Unikalny ciąg znaków w postaci łącza internetowego, które odnosi się do abstraktu lub zasobu fizycznego . Te identyfikatory są używane do lokalizowania zasobów (w tym węzłów IoT) i interakcji z nimi w sieciach przy użyciu przyjaznych dla człowieka nazw strukturalnych, które opisują ich przeznaczenie.

* Elektroniczny kod produktu (EPC)9. Globalny standard tagów EPC, który przypisuje niepowtarzalną tożsamość każdemu obiektowi fizycznemu w dowolnym momencie na całym świecie (np. EPC = urn: epc:id:sgtin:0614141.012345.62852). EPC są kodowane na pasywnych znacznikach RFID, które mogą być używane do śledzenia wszelkiego rodzaju obiektów, takich jak pojazdy na płatnych drogach.

* Wszechobecny kod (uCode)10. Numeryczny system identyfikacji, który jednoznacznie identyfikuje fizyczne obiekty i miejsca w rzeczywistym świecie. Używając uCode, informacje mogą być powiązane z obiektami i lokalizacjami i mogą być pobierane z www.uidcenter.org. Przykładowy uCode to 00001C0000000000000001000285E7A6E3. W przeciwieństwie do EPC, uCode oferuje większą elastyczność, ponieważ tagi mogą przybierać różne formy, takie jak drukowanie (tj. QR i kody kreskowe), pasywny RFID, aktywny RFID, tagi na podczerwień i akustyczne.

Techniki adresowania umożliwiają przypisanie unikalnych adresów sieciowych do węzłów w celu lepszego zarządzania siecią. Protokoły adresowania są następujące:

* IPv4. Popularny protokół sieciowy wykorzystujący 32-bitowe adresy do przypisywania węzłów do unikalnych adresów i agregowania ich w mniejsze sieci. Chociaż IPv4 jest szeroko stosowany, nie jest zalecany dla IoT ze względu na niewielką liczbę adresów (tj. 232 adresy), które można wykorzystać.

* IPv6. Protokół sieciowy, którego celem jest przezwyciężenie ograniczeń IPv4 poprzez obsługę 2128 adresów, dzięki czemu każde urządzenie może mieć unikalny adres. W przeciwieństwie do protokołu IPv4, protokół IPv6 zapewnia automatyczną konfigurację, która umożliwia przypisywanie urządzeniom adresów IP bez potrzeby korzystania z serwera protokołu DHCP (Dynamic Host Configuration Protocol). Autokonfiguracja jest ważną funkcją IoT, ponieważ upraszcza wdrażanie WSN na dużą skalę. Ze względu na duży rozmiar nagłówka (tj. 320 bitów) w IPv6, IPv6 przez bezprzewodowe sieci osobiste o niskim poborze mocy (6LoWPAN) jest używany do kompresji adresu, aby był kompatybilny z tradycyjnymi protokołami WSN (tj. IEEE 802.15.4), jako zostanie później opisany.

Komunikacja

Po zidentyfikowaniu węzła (tj. zaadresowaniu i nazwaniu) może rozpocząć komunikację z innymi węzłami lub serwerami zaplecza. Jednak komunikacja ta wymaga wyboru odpowiedniego medium komunikacyjnego, które zależy od możliwości węzła (np. moc i zasięg). Węzły mogą komunikować się poziomo (tj. ad hoc z innym węzłem) lub pionowo (tj. z serwerami w warstwie oprogramowania pośredniego). W tej sekcji omówiono technologie komunikacyjne z naciskiem na ich zastosowanie w IoT. Te powszechnie stosowane technologie są wymienione w porządku rosnącym na podstawie ich zasięgu bezprzewodowego (tj. przewodowego oraz krótkiego, średniego i dalekiego zasięgu):

* Komunikacja przez linię elektroenergetyczną (PLC). Obejmuje to zestaw protokołów komunikacyjnych, które wykorzystują okablowanie linii energetycznej do jednoczesnego przesyłania zarówno danych, jak i prądu przemiennego. Stosując to podejście, osoba może zasilać urządzenia i kontrolować/odzyskiwać dane za pomocą tylko standardowych kabli zasilających, które biegną do urządzenia. PLC może być używany do łączenia węzłów w środowisku IoT w celu udostępniania danych zapleczu, które następnie wykonuje pewne działania. Jest to preferowane głównie w węzłach stacjonarnych, ponieważ i tak opierają się na liniach energetycznych, a także w środowiskach IoT, które charakteryzują się wysokimi zakłóceniami bezprzewodowymi w inteligentnych pojazdach, inteligentnych sieciach i inteligentnych domach.

* X10. Podobnie jak PLC, X10 jest standardem przemysłowym, który wykorzystuje okablowanie elektryczne do sygnalizacji i sterowania urządzeniami, w których sygnały zawierają krótkie impulsy o częstotliwości radiowej (RF), które mogą zawierać dane. Pomimo niższej szybkości transmisji danych i zasięgu w porównaniu z PLC, pozostaje popularnym wyborem ze względu na niski koszt. X10 jest stosowany przede wszystkim do łączenia węzłów w inteligentnych domach, w których moduły gniazd (np. włączniki światła, wyłączniki AC itp.) mogą być sterowane z aplikacji mobilnej lub portalu internetowego.

* Komunikacja bliskiego pola (NFC). Zestaw protokołów umożliwiający komunikację między dwoma urządzeniami na bardzo krótkie odległości. NFC w przeciwieństwie do RFID, który służy przede wszystkim do identyfikacji, oferuje prosty sposób uwierzytelniania, uzyskiwania dostępu i udostępniania danych między urządzeniami i użytkownikami w systemach IoT. NFC jest wykorzystywane w wielu aplikacjach IoT, takich jak inteligentne płatności i systemy kontroli dostępu.

* Ultraszeroka przepustowość (UWB). Starsza technologia bezprzewodowa, która wykorzystuje transmisje niskoenergetyczne w celu zapewnienia komunikacji o wysokiej przepustowości w szerokim spektrum radiowym. Ta technologia jest odpowiednia dla IoT ze względu na odporność na zakłócenia, co czyni ją wysoce skalowalną, ponieważ dodanie większej liczby węzłów nie spowoduje zakłóceń. Ta technologia może pomóc sieciom bezprzewodowym stać się bardziej adaptacyjne i wydajne, ponieważ każdy węzeł może być śledzony za pomocą UWB. UWB jest zwykle wykorzystywany w usługach lokalizacyjnych (np. śledzenie zasobów).

* Wi-Fi. Wi-Fi jest szczególnie przydatne w konfiguracjach ad hoc, takich jak Wi-Fi Direct, które nie wymagają bezprzewodowego punktu dostępowego. Głównym ograniczeniem Wi-Fi jest zużycie energii. Jednak w niektórych aplikacjach IoT (np. w inteligentnych domach) moc nie jest istotną kwestią. Organizacja Wi-Fi Alliance wprowadza na rynek nową, energooszczędną technologię Wi-Fi o nazwie Wi-Fi HaLow, zaprojektowaną specjalnie dla węzłów IoT.

* IEEE 802.15.413. Standard dla bezprzewodowych sieci osobistych o niskiej przepustowości (LR-WPAN), który określa warstwę fizyczną i kontrolę dostępu do mediów. Ponieważ IEEE 802.15.4 został zaprojektowany dla IPv4, większe datagramy IPv6 nie pasują naturalnie do sieci IEEE 802.15.4. Rozwiązaniem tego problemu jest zastosowanie warstwy adaptacyjnej (tj. 6LoWPAN), która hermetyzuje i kompresuje nagłówki, aby umożliwić przesyłanie pakietów IPv6 przez sieci IEEE 802.15.4. ZigBee, ISA100.11a, WirelessHART, MiWi i Thread są oparte na IEEE 802.15.4 i różnią się tylko górnymi warstwami. ZigBee i Thread są powszechnie stosowane w inteligentnych domach i inteligentnych licznikach, podczas gdy ISA100.11a, MiWi i WirelessHART są powszechnie używane w zastosowaniach przemysłowych.

* Bluetooth Low Energy (BLE). Standard bezprzewodowy jest przeznaczony do wymiany danych na niewielkie odległości i budowania osobistych sieci komputerowych (PAN). W przypadku Bluetooth w wersji 4.0 i nowszych poprawiono zużycie energii, co sprawia, że Bluetooth jest dobrze przystosowany do czujników i innych małych urządzeń, które wymagają małej mocy. Węzły z dostępem do Internetu często używają BLE do działania z węzłami lokalnymi i wysyłania zebranych danych do zaplecza w celu wykonania większej liczby działań. BLE może być używany w szerokim zakresie zastosowań, takich jak inteligentne budynki, inteligentny transport i urządzenia do noszenia.

* ANT+. Zastrzeżony (ale o otwartym dostępie) stos protokołów komunikacji bezprzewodowej umożliwia węzłom komunikację poprzez tworzenie reguł współistnienia, sygnalizacji, reprezentacji danych, uwierzytelniania i wykrywania błędów. Jest on stosowany przede wszystkim w aplikacjach IoT związanych ze sportem, dobrym samopoczuciem i monitorowaniem stanu zdrowia domu, w których każdy węzeł IoT wykorzystuje lub wysyła wykryte dane do chmury w celu wykonania większej liczby działań.

* Z-Wave. Technologia szeroko stosowana w inteligentnych domach. Urządzenia Z-Wave można podłączać do sprzętu AGD, co umożliwia sterowanie nimi przez Internet.

* Nieważkość. Zestaw otwartych standardów technologii bezprzewodowej o niskim poborze mocy (LPWAN) służących do wymiany danych między stacją bazową a tysiącami okolicznych węzłów. Istnieją trzy różne standardy dla nieważkości, a mianowicie nieważkość-W, nieważkość-P i nieważkość-N; z których każdy ma inne przypadki użycia ze względu na swoje cechy. Weightless-W oferuje dwukierunkową komunikację, która jest idealna do zastosowania w inteligentnym sektorze ropy i gazu, gdzie prawdopodobnie dostępna będzie biała przestrzeń TV (TVWS). Chociaż weightless-P zapewnia również komunikację dwukierunkową, jest idealny do sieci prywatnych i sytuacji, które wymagają zarówno funkcji uplink, jak i downlink. Weightless-N oferuje łącze nadrzędne, które czyni go idealnym rozwiązaniem dla sieci opartych na czujnikach, takich jak odczyty temperatury, monitorowanie poziomu w zbiorniku i pomiary.

* Sieci komórkowe. Sieci te istnieją w różnych generacjach (np. 3G i 4G) standardów sieci komórkowych, które są często stosowane w smartfonach, ale są również odpowiednie dla IoT ze względu na ich wysoką mobilność i wyższą prędkość. W ostatnich pokoleniach poczyniono znaczne postępy w zakresie zużycia energii i szybkości. Na przykład najnowsza specyfikacja 4G (zwana LTE-Advanced) oferuje tryb oszczędzania energii, który wydłuża żywotność baterii urządzenia do dekady z

maksymalną prędkością 10 Mb/s. Dla porównania, inne wersje 4G (mniej energooszczędne) mogą mieć prędkość do 1 Gb/s.

* SigFox. Usługa oparta na subskrypcji, która oferuje rozwiązania łączności (w niektórych krajach) za pośrednictwem dedykowanych sieci LPWAN. SigFox jest idealnym rozwiązaniem dla aplikacji IoT, które muszą wysyłać rzadkie i małe porcje danych, takich jak systemy alarmowe lub inteligentne liczniki.

* Protokół sojuszu DASH7 (D7A). Bezprzewodowy protokół sieciowy czujnika i siłownika typu open source, który zapewnia wieloletnią żywotność baterii i niskie opóźnienia w łączeniu ruchomych węzłów. Jego zastosowania IoT obejmują inteligentne budynki, usługi lokalizacyjne, logistykę i inteligentne pojazdy.

* Sieć rozległa dalekiego zasięgu (LoRaWAN). Technologia firmy LoRa Alliance, która oferuje niedrogą, mobilną i bezpieczną komunikację dwukierunkową. Jest zoptymalizowany pod kątem niskiego zużycia energii i przeznaczony do obsługi dużych sieci z milionami węzłów, co czyni go idealnym do zastosowań IoT, takich jak inteligentne miasta i aplikacje przemysłowe. Niektóre dodatkowe technologie, które mają zostać przyjęte w aplikacjach IoT, są obecnie w fazie rozwoju i prawdopodobnie będą używane w ciągu najbliższych kilku lat, ponieważ rozwiążą niektóre ograniczenia komunikacyjne w obecnych technologiach. Obejmuje to na przykład:

* 5G. Kandydat do komunikacji średniego zasięgu, który powinien być dostępny do 2020 roku. Oferuje wiele ulepszeń w porównaniu z 4G, takich jak zwiększona prędkość (prawie odpowiednik Ethernetu) i lepszy zasięg oraz obsługuje dużą liczbę użytkowników. Tym samym umożliwi podłączenie większej liczby węzłów z dodatkową mobilnością.

* Wierność światła (Li-Fi). Nowa technologia bezprzewodowa oferująca szybką transmisję danych z wykorzystaniem komunikacji w świetle widzialnym (VLC). Li-Fi może również wykorzystywać energooszczędne diody elektroluminescencyjne (LED), aby obniżyć koszty energii. Jednak schemat Li-Fi jest ograniczony do linii wzroku i podlega zakłóceniom lub uszkodzeniom z powodu obiektów blokujących sygnały świetlne. Chociaż Li-Fi nie jest szeroko stosowane w IoT, jego duża prędkość wskazuje, że ma znaczny potencjał do wykorzystania w wielu zastosowaniach, takich jak inteligentne teatry i inteligentne sale lekcyjne.

* Sieć definiowana programowo (SDN). Architektura sieciowa, która jest dynamiczna, łatwa w zarządzaniu, elastyczna i opłacalna. Te cechy sprawiają, że sieci SDN są bardzo skuteczne, biorąc pod uwagę dużą przepustowość i dynamiczną naturę Internetu Rzeczy. Niewykorzystane zasoby sieciowe i inteligentne przemieszczanie tras ułatwią planowanie zalewu danych, którego oczekuje się wraz ze wzrostem wykorzystania Internetu Rzeczy. Sieci SDN minimalizują wąskie gardła i zapewniają wydajność sieci

Każda z tych obecnych technologii komunikacyjnych ma specjalne cechy, które umożliwiają efektywne wdrożenie różnych scenariuszy. Szybkość transmisji danych pomaga w wyborze najlepszej technologii w oparciu o wymagania aplikacji dotyczące przepływu danych. Podobnie zasięg jest ważnym czynnikiem dla węzłów IoT pod względem mobilności. Bezpieczeństwo jest również istotną cechą, ponieważ czujniki na dużą skalę mogą spowodować katastrofalne uszkodzenia, gdy zostaną naruszone. Dlatego technologie, które nie oferują żadnych wbudowanych zabezpieczeń, powinny być stosowane tylko z dodatkowymi protokołami zabezpieczeń zapewnianymi przez inne warstwy, jak omówiono w następnej sekcji. Komunikacja ad hoc ma kluczowe znaczenie w aplikacjach, w których węzły komunikują się ze sobą bez kontrolera. Technologie uważane za natywne protokoły TCP/IP można łatwo zintegrować z obecnymi systemami. Rozmiar nagłówka może wpływać na wybór technologii o niskich szybkościach transmisji danych. Opóźnienia mogą być ważne w niektórych zastosowaniach IoT

(np. w chirurgii zdalnej. Całkowite opóźnienie często wzrasta, gdy wdrażana jest duża liczba czujników. Częstotliwość jest ważna, aby uniknąć zakłóceń sygnału lub ograniczeń w niektórych miejscach. Skalowalność jest również ważną cechą IoT Technologie, które są skalowalne, często pozwalają na dodanie nowej dużej liczby węzłów bez wpływu na wydajność, podczas gdy technologie nieskalowalne są zazwyczaj ograniczone do mniej niż 100 węzłów na sieć.

Bezpieczeństwo

Ze względu na dużą liczbę węzłów o ograniczonych możliwościach bezpieczeństwo jest ważnym i trudnym zadaniem, ponieważ udany atak może spowodować nadmierne szkody (np. ataki typu Distributed Denial-of-Service (DDoS)). Jak omówiono w sekcji o technologiach komunikacyjnych, zabezpieczenia mogą nie być wbudowane w różne technologie komunikacyjne (np. Z-wave i Sigfox). Dlatego dodatkowe mechanizmy bezpieczeństwa muszą być zapewnione przez różne warstwy, aby zmniejszyć prawdopodobieństwo ataków. W tej warstwie mogą wystąpić ataki z powodu niezabezpieczonej komunikacji między węzłami. Dlatego do bezpiecznej komunikacji wymagane są lekkie mechanizmy bezpieczeństwa. Niektóre sugerowane techniki bezpieczeństwa są następujące:

* Zabezpieczenia protokołu internetowego (IPsec)27. Dobrze znany protokół warstwy sieciowej używany z protokołem IPv6 do uwierzytelniania i szyfrowania typu end-to-end między węzłami. Ponieważ IPsec jest zaimplementowany w warstwie sieciowej (w TCP/IP), obsługuje również warstwy wyższe. Możliwa jest lekka implementacja protokołu IPsec, co czyni go idealnym dla węzłów IoT.

* Transport Layer Security (TLS)28 i Datagram TLS (DTLS. Są to dobrze znane protokoły kryptograficzne, które są również używane odpowiednio z protokołami TCP i protokołami datagramów użytkownika (UDP) w celu zapewnienia bezpiecznej komunikacji.

* IEEE 188830. Ten protokół jest częścią rodziny standardów dla Ubiquitous Green Community Control Network. IEEE 1888.3 zapobiega nieautoryzowanemu dostępowi do zasobów i przypadkowym wyciekom danych, jednocześnie zwiększając poufność i integralność przesyłanych danych.

Wytyczanie

Gdy węzeł zna adres docelowy, musi mieć wydajną metodę trasowania danych do tego węzła docelowego. Wydajny routing ma kluczowe znaczenie w środowiskach IoT ze względu na ogromną liczbę węzłów, które można połączyć ad hoc. Ze względu na ograniczone możliwości węzłów IoT, każdy węzeł musi sprawnie poznać optymalną trasę. Dlatego dla tych środowisk wprowadzono specjalny protokół routingu. Routing Protocol for Low Power and Lossy Networks (RPL) to standardowy protokół routingu IPv6 dla urządzeń o ograniczonych zasobach. Protokół ten został zaproponowany do obsługi różnych typów łączy, takich jak IEEE 802.15.4, oraz popularnych typów ruchu, w tym jeden-do-jednego, wiele-do-jednego i jeden-do-wielu. Protokół można zwięźle przedstawić jako zestaw wykresów, w których każdy wykres jest ukierunkowanym na cel acyklicznym wykresem (DODAG). W tego typu grafie każdy węzeł zna co najmniej jedną ścieżkę do swojego węzła głównego (tj. routera granicznego), a każdy węzeł jest świadomy swojego rodzica. Węzły wymieniają specjalne komunikaty RPL w celu utrzymania wykresu i zapewnienia, że poprawna trasa (trasy) do katalogu głównego jest zawsze dostępna. Routery mogą pracować w dwóch trybach operacyjnych: trybie bez pamięci (jak w DODAG 1) oraz w trybie z pamięcią (jak w DODAG2). W trybie bez przechowywania węzły przekazują każdą wiadomość do korzenia, co może być przydatne w scenariuszach, w których węzły są zaprogramowane do okresowego wysyłania wykrytych danych (np. odczytów temperatury). W przeciwieństwie do tego, w trybie przechowywania komunikacja jest dwukierunkowa, a każdy węzeł przechowuje trasy do wszystkich znajdujących się pod nim węzłów. Może to być pomocne w scenariuszach, w których węzły

wysyłają wykryte dane tylko wtedy, gdy są one wymagane; stąd potrzebne są trasy w dół, aby dostarczyć komunikaty żądań.

Technologie oprogramowania pośredniego

Warstwa oprogramowania pośredniego (znana również jako warstwa zarządzania usługami) jest rdzeniem środowiska IoT. Może być mapowany do warstwy aplikacji w pakiecie IP (TCP/IP). Technologie w tej warstwie są często wspierane przez platformy IoT. Ta warstwa umożliwia identyfikację i żądanie usług na podstawie nazw i adresów oraz umożliwia programistom interakcję z heterogenicznymi obiektami, niezależnie od konkretnej konfiguracji sprzętowej. Warstwa ta również przetwarza otrzymane dane, podejmuje decyzje i dostarcza wymagane usługi. Warstwa oprogramowania pośredniczącego realizuje te zadania za pomocą poniższych technologii, które zostały podzielone na trzy grupy na podstawie ich funkcjonalności.

Wykrywanie usług

Aplikacje IoT i użytkownicy muszą mieć możliwość używania nazw lub adresów do żądania usług bez znajomości szczegółów infrastruktury bazowej. Dlatego usługi muszą być wykrywalne i zarejestrowane. Skalowalne i heterogeniczne środowiska, rejestracja usług i wykrywanie powinny być autonomiczne, dynamiczne i wydajne. Protokoły najczęściej używane do wykrywania usług to:

* Multiemisja DNS (mDNS). Protokół o zerowej konfiguracji i niezależny od infrastruktury, który może nadal działać nawet podczas awarii infrastruktury, ponieważ może korzystać z lokalnych pamięci podręcznych. Wiąże nazwy hostów z adresami IP w małych sieciach, rozsyłając wiadomość z żądaniem IP, prosząc urządzenie o określonej nazwie hosta o wysłanie swojego adresu IP. Ten host odpowiada wiadomością multiemisji zawierającą jego adres IP. Wszystkie inne urządzenia w podsieci używają tej odpowiedzi do aktualizowania lokalnych pamięci podręcznych mDNS. Protokół ten, w połączeniu z IPv6 (z automatyczną konfiguracją), może całkowicie wyeliminować potrzebę lokalnego serwera do zarządzania lokalnymi WSN.

* Wykrywanie usług DNS (DNS-SD). Kolejny protokół o zerowej konfiguracji, który uzupełnia mDNS, ponieważ łączy usługi z hostami w domenie. Klienci IoT mogą znaleźć usługę, wysyłając żądanie multiemisji dla typu usługi (np. odczyty temperatury), a wszystkie maszyny, które świadczą tę usługę, odpowiedzą swoimi nazwami hostów. Nazwy są ważne, ponieważ nazwy węzłów, w przeciwieństwie do adresów IP, rzadko się zmieniają. Następnie mDNS jest używany do parowania nazw węzłów z ich adresami IP. Avahi i Bonjour to dwie popularne implementacje zarówno mDNS, jak i DNS-SD.

Innym podejściem do wykrywania usług jest publikowanie jednolitych identyfikatorów zasobów (URI) węzłów, które świadczą usługi w sieci Web. Takie podejście jest znane jako sieć rzeczy (WoT). Technologie, które przyjmują to podejście, to:

* HyperCat. Otwarty, lekki format katalogu hipermedialnego oparty na JavaScript Object Notation (JSON) jest przeznaczony do publikowania kolekcji identyfikatorów URI. Każdy katalog HyperCat może zawierać dowolną liczbę identyfikatorów URI, każdy z dowolną liczbą potrójnych instrukcji typu RDF (Resource Description Format), które dostarczają informacji o tym identyfikatorze URI. HyperCat umożliwia programistom publikowanie połączonych opisów danych węzłów IoT, które mogą być pobierane przez żądania HTTP GET, co wymaga od klientów IoT znajomości adresu URI.

* Internet fizyczny. Technologia, która umożliwia węzłom rozgłaszanie ich adresów URL (Uniform Resource Locator) lub niektórych danych lokalnych na stronach internetowych. W przeciwieństwie do HyperCat, klienci IoT nie muszą znać adresów URL, ponieważ mogą je pobierać za pomocą sygnałów nawigacyjnych Bluetooth (np. Eddystone), mDNS lub Wi-Fi Direct. Opublikowane adresy URL

umożliwiają użytkownikom interakcję z węzłami (np. czujnikami) ze smartfonów lub laptopów. Usługa proxy służy do zapobiegania złośliwym adresom URL i sortowania list adresów URL na podstawie preferencji użytkowników.

* Uniwersalny Plug and Play (UPnP). Grupa protokołów sieciowych, które umożliwiają sieciowym węzłom IoT bezproblemowe wykrywanie wzajemnej obecności w sieci i dostarczanie funkcjonalnych usług sieciowych (tj. udostępniania danych, komunikacji i rozrywki). Wiadomo jednak, że UPnP jest podatny na problemy z bezpieczeństwem. UPnP+ to nowa inicjatywa, która ma na celu rozwiązanie problemów związanych z UPnP.

Wymiana danych

Aby zarządzać ogromną ilością danych tworzonych przez dużą liczbę węzłów, menedżer musi przesyłać dane do odbiorców w bezpieczny i wydajny sposób. To zarządzanie odbywa się w warstwie aplikacji TCP/IP. Tak więc technologie, które się tutaj znajdują, są znane jako protokoły aplikacji. W środowiskach IoT każdy węzeł może przechwytywać dane, analizować dane i kontrolować inne węzły. Jednak zanim węzeł będzie mógł rozpocząć komunikację i wymianę danych w sieci IoT, musi zostać zidentyfikowany i zarejestrowany jako część sieci usługowej. Ta operacja jest nazywana subskrypcją węzła. Węzeł subskrybujący musi złożyć żądanie przyłączenia się do sieci usługowej (tj. subskrybować), zanim będzie mógł uzyskać lub opublikować dane związane z usługą (np. temperaturę) z lub do sieci. Gdy węzeł zostanie zasubskrybowany dla wydawców, po opublikowaniu otrzyma dane związane z usługami. Ten model jest znany jako model publikowania/subskrypcji. Alternatywne podejście nazywa się żądaniem/odpowiedzią, w którym dołączony węzeł zawsze żąda jawnie danych związanych z usługą i otrzymuje odpowiedź zawierającą żądane dane. Gdy węzeł dołączy do sieci, przydzielane są mu zasoby sieciowe i może rozpocząć wymianę danych oraz działać jako komponent środowiska IoT. Model publikowania/subskrypcji jest generalnie preferowany w scenariuszach IoT, które wymagają dużego wdrożenia węzłów, które często wymieniają dane. Rzeczywiście, w takich scenariuszach korzystanie z modelu żądanie/odpowiedź prowadzi do dużego obciążenia sieci, co może powodować zwiększone zużycie energii. W przeciwieństwie do tego model żądanie/odpowiedź można łatwo zintegrować z aplikacjami klient/serwer, co zostanie omówione w przypadku protokołu ograniczonej aplikacji (CoAP). Niektóre z powszechnie używanych protokołów to:

* Protokół CoAP.Arequest/response został stworzony przez grupę Constrained RESTful Environments (CoRE) dla urządzeń z ograniczeniami. Aby zapewnić interakcje zorientowane na zasoby w architekturze żądania/odpowiedzi, używa poleceń HTTP (tj. GET, POST, PUT i DELETE) przez UDP. W przeciwieństwie do TCP, UDP nadaje się do lekkich implementacji IoT. Serwery proxy HTTP-CoAP mogą być używane do łatwej konwersji interakcji między HTTP i CoAP. Aby zapewnić niezawodność, której brakuje UDP, nagłówek wiadomości CoAP ma dwa bity, które określają wymagany poziom jakości usługi (QoS). CoAP obejmuje cztery rodzaje wiadomości: potwierdzalne (tzn. muszą być potwierdzone przez odbiorcę pakietem potwierdzenia [ACK]), niepotwierdzalne (tzn. komunikaty „odpal i zapomnij”, które nie wymagają potwierdzenia), potwierdzające (tzn. potwierdzenie komunikat możliwy do potwierdzenia) i zresetuj (tj. odrzucenie wiadomości lub usunięcie obserwatora). CoAP ma prosty mechanizm retransmisji back-off w celu wykrywania i zapobiegania duplikatom dla potwierdzalnych wiadomości za pomocą unikalnego identyfikatora wiadomości, który jest 16-bitowym polem nagłówka. DTLS może być używany do zapewnienia bezpieczeństwa aplikacjom CoAP; jednak DTLS ma problemy, ponieważ (1) jego uzgadnianie powoduje, że węzły o ograniczonych zasobach zużywają dodatkowe zasoby, co prowadzi do skrócenia żywotności baterii; (2) zwiększa złożoność konwersji między HTTP a CoAP; oraz (3) uniemożliwia obsługę multiemisji CoAP.

* Rozszerzalny protokół przesyłania wiadomości i obecności (XMPP). Protokół IETF do komunikacji przez Internet. Jest odpowiedni dla IoT, ponieważ zapewnia komunikację niemal w czasie rzeczywistym, z małymi opóźnieniami i niezależną od platformy w sposób zdecentralizowany. Działa przez TCP i działa zarówno w modelu publikowania/subskrypcji, jak i żądania/odpowiedzi. W XMPP klient łączy się z serwerem za pomocą komunikatów eXtensible Markup Language (XML). Parsowanie XML powoduje duże obciążenie sieci; dostępne są jednak metody zmniejszenia tego obciążenia (Waher i DOI, 2015). XMPP ma wbudowane podstawowe funkcje bezpieczeństwa TLS/Secure Sockets Layer (SSL). QoS nie jest dostępny w XMPP; dziedziczy mechanizmy TCP zapewniające niezawodność, które nie są odpowiednie dla IoT. Aby rozwiązać ten problem, projekt normy (tj. XEP-0184: zaproponowano potwierdzenie odbioru wiadomości

* Transport telemetryczny kolejki wiadomości (MQTT). Lekki protokół komunikacyjny M2M, który wykorzystuje model publikowania/subskrybowania. Nadaje się do powolnych, zawodnych połączeń i urządzeń o ograniczonej mocy obliczeniowej. MQTT zawiera trzy rodzaje modułów: broker, wydawcy i subskrybenci. Abonent może zarejestrować się do określonej usługi; następnie broker powiadamia zainteresowanych subskrybentów, gdy wydawcy uruchamiają interesującą ich usługę. MQTT działa na szczycie TCP i zapewnia trzy poziomy QoS: co najwyżej raz (tj. wiadomość zostanie dostarczona raz bez potwierdzenia), co najmniej raz (tj. wiadomość zostanie dostarczona co najmniej raz, z wymaganym potwierdzeniem), i dokładnie raz (tj. wiadomość zostanie dostarczona dokładnie raz przy użyciu czteroetapowego uścisku dłoni). Nieco inna wersja MQTT dla sieci czujników (tj. MQTT-SN) jest skierowana do urządzeń wbudowanych na innych Sieci TCP/IP, takie jak ZigBee. Bezpieczeństwo realizowane jest przez brokera MQTT za pomocą protokołu TLS/SSL, co wymaga uwierzytelnienia subskrybenta.

* Zaawansowany protokół kolejkowania wiadomości (AMQP). Otwarty standard IoT, który zapewnia zorientowane na wiadomości możliwości komunikacji publikowania/subskrybowania. Podobnie jak w przypadku MQTT, niezawodność wiadomości jest utrzymywana przy użyciu gwarancji dostarczania wiadomości, takich jak co najmniej raz, co najmniej raz i dokładnie raz. TLS/SSL i/lub Simple Authentication and Security Layer (SASL)45 są używane do zabezpieczenia protokołu TCP. Komunikacja w AMQP jest zarządzana przez dwa składniki: wymiany i kolejki komunikatów. Wymiany kierują wiadomości do odpowiednich kolejek lub między nimi w oparciu o predefiniowane reguły. Kolejki wiadomości przechowują wiadomości w kolejkach, zanim zostaną wysłane do odbiorców. Chociaż brakuje obsługi protokołu dla kolejek ostatniej wartości (LVQ)46, AMQP zapewnia bogatszą grupę scenariuszy przesyłania wiadomości (np. jeden do jednego, emisja, filtrowanie tematów i odpowiedź na żądanie)

* Usługa dystrybucji danych (DDS)47. Standard M2M stworzony przez Object Management Group (OMG). Jest to zorientowany na dane protokół publikowania/subskrypcji, który nie opiera się na brokerze. Ze względu na obsługę multiemisji jest to bardzo niezawodny protokół, ponieważ oferuje 23 różne polityki QoS. Architektura DDS składa się z dwóch poziomów interfejsu. Pierwszy poziom – Data-Centric Publish-Subscribe (DCPS) - odpowiada za dostarczanie informacji subskrybentom. Drugi poziom - warstwa rekonstrukcji danych lokalnych (DLRL) - to opcjonalna warstwa, która dzieli obiekty danych na oddzielne elementy, aby umożliwić prostą integrację DDS z warstwą aplikacji. Oprócz wydawców i subskrybentów typowa aplikacja DDS zawiera następujące elementy: (1) domena (np. urządzenia medyczne), (2) temat (tj. grupa danych z podobnych urządzeń, np. odczyty elektrokardiogramu (EKG) z wielu urządzeń), (3) instancja (tj. cel, w którym dane się zmieniają, np. „Odczyty EKG pacjenta #13”), (4) próbka (tj. migawka instancji w punkt w czasie), (5) Data-Writer (tj. źródło informacji na dany temat) oraz (6) DataReader (tj. obserwator tematu).

Obliczenie

Obliczenia są wymagane do przetwarzania zebranych danych i zarządzania czujnikami. Obliczenia w środowiskach IoT mogą mieć zakres od lekkich do złożonych operacji. Lekkie obliczenia są związane z zarządzaniem platformą, takie jak listy dostępu do usług, które obejmują szyfrowanie/deszyfrowanie. Złożone obliczenia dotyczą rzeczywistych danych (np. logiki). Obliczenia można wykonywać w oparciu o wymagania aplikacji IoT w następujących lokalizacjach:

* Lokalny. Obliczenia lokalne są wykonywane przy użyciu jednostki przetwarzającej lub systemu na chipie (SoC). Aplikacje mogą obejmować technologie sieciowe i jednopłytkowe, takie jak Arduino, UDOO, produkty Intel IoT i Raspberry PI. Ten sprzęt jest zwykle obsługiwany przez systemy operacyjne czasu rzeczywistego (RTOS), takie jak Contiki OS52 i RIOT OS. Ten typ może być używany w małych projektach IoT (np. systemy bezpieczeństwa typu „zrób to sam" (DIY) lub w routerach granicznych do lekkich operacji, takich jak filtrowanie i kompresowanie danych, w celu zmniejszenia obciążenia sieci i obciążenia serwerów w chmurze lub mgle .

* Chmura. Przetwarzanie w chmurze jest preferowane w przypadku aplikacji, w których czujniki znajdujące się w różnych miejscach wysyłają wygenerowane dane do chmury w celu scentralizowanego przetwarzania. Takie podejście jest elastyczne, ponieważ usługi w chmurze można skalować w górę lub w dół na żądanie. Oparte na chmurze platformy IoT są najpopularniejszą architekturą w obecnych rzeczywistych wdrożeniach IoT. Kaa i DeviceHive to przykłady wdrożeń opartych na chmurze IoT Platform as a Service (PaaS). Cloud Infrastructure as a Service (IaaS) jest używany przez dostawców usług IoT, aby umożliwić swoim klientom korzystanie z usług IoT (tj. oprogramowania jako usługi (SaaS)).

* Mgła. Warstwa przetwarzania mgły jest idealnie stosowana w chmurze w celu poprawy wydajności, ponieważ może być wdrażana w pobliżu użytkowników końcowych lub węzłów. Jest to ważne, ponieważ serwery w chmurze mogą znajdować się daleko od węzłów; w ten sposób serwery mgły obliczeniowej mogą wykonywać pewne operacje, aby zmniejszyć opóźnienie przesyłania danych do chmury. Ponadto mgła jest bardziej preferowana niż chmura, ponieważ tylko niewielka ilość przesyłanych danych jest istotna dla praktycznych wglądów. Dlatego chmura nie zawsze jest wydajna. Tysiące jednostek obliczeniowych mgły są dostarczane przez operatorów sieci komórkowych, ale nie mają takich samych możliwości obliczeniowych jak chmura. Podobnie jak w przypadku przetwarzania w chmurze, przetwarzanie we mgle poprawia skalowalność. Niskie opóźnienia i świadomość lokalizacji, powszechna dystrybucja graficzna, gęstość urządzeń, mobilność i możliwości w czasie rzeczywistym to zalety obliczeń mgły

Technologie warstwy aplikacji

Ta warstwa jest odpowiedzialna za dostarczanie żądanych usług użytkownikom IoT za pośrednictwem prostego interfejsu bez znajomości sposobu przetwarzania żądań usług w warstwach leżących poniżej. Użytkownicy IoT mogą uzyskać dostęp do usługi (np. zdalne odczytywanie lub ustawianie warunków temperaturowych lub śledzenie pojazdów i zarządzanie nimi) za pomocą wielu platform (np. laptopów, smartfonów i smartwatchów) za pośrednictwem portali internetowych lub aplikacji. Usługi różnią się w zależności od scenariusza IoT, ale można je podzielić na cztery główne klasy, jak opisano w poniższych sekcjach.

Usługi związane z tożsamością

Usługi związane z tożsamością wymagają identyfikatora osadzonego w węźle i czytniku, takim jak urządzenie RFID. Usługi związane z tożsamością mogą być aktywne lub pasywne (jak omówiono w podrozdziałach RFID). Usługi te są niezwykle ważne w aplikacjach IoT, ponieważ śledzą urządzenia w

dużych wdrożeniach (np. rejestrują konserwację urządzeń i inwentaryzacje zużytych części i stanów magazynowych). Przykładem tego typu usługi jest aplikacja do śledzenia przesyłek.

Usługi agregacji informacji

Usługi agregacji podsumowują zebrane surowe pomiary sensoryczne z różnych typów czujników i sieci, które muszą być przetwarzane i zgłaszane do aplikacji IoT. Przykładem tego typu usługi jest rozdział obciążenia między inteligentne sieci.

Usługi oparte na współpracy

Usługi świadome współpracy są oparte na usługach agregacji informacji i służą do podejmowania decyzji dotyczących pozyskanych danych. Tego typu usługi można znaleźć między innymi w inteligentnych domach, inteligentnym rolnictwie i inteligentnej produkcji. Na przykład w inteligentnym domu system bezpieczeństwa i inteligentne termostaty są stosowane razem, aby poprawić bezpieczeństwo i efektywność energetyczną w oparciu o wymianę danych między nimi.

Wszechobecne usługi

Są to najkorzystniejsze usługi IoT, ponieważ przenoszą usługi świadome współpracy na wyższy poziom, oferując pełny dostęp do wszystkiego w (prawie) dowolnym czasie i z dowolnego miejsca. Dostęp i kontrolę można uzyskać za pomocą komputera, smartfona lub dowolnego urządzenia inteligentnego. Przykładami takich usług są inteligentne miasta.

Technologie warstwy biznesowej

W tej warstwie, w przeciwieństwie do warstwy aplikacji, można uzyskać dostęp do danych usług i danych środowiskowych IoT, takich jak modele biznesowe, schematy blokowe i wykresy. Dostęp ten pomaga administratorom w projektowaniu, analizie, wdrażaniu, ocenie, monitorowaniu i rozwoju systemów IoT, ponieważ dane wyjściowe każdej z wcześniej wymienionych warstw są analizowane w tej warstwie w celu poprawy usług i ochrony prywatności użytkowników. Technologie w tej warstwie mogą być podzielone na dwie kategorie w zależności od ich funkcji (tj. semantyki i analityki Big Data), jak opisano w kolejnych częściach.

Semantyka

Po przechwyceniu danych przez czujniki można je analizować w celu wydobycia wiedzy. Wiedza ma kluczowe znaczenie dla ulepszania usług i uzyskiwania użytecznych wyników, których nie można odkryć bez zbadania wszystkich danych środowiska IoT. Aby skutecznie analizować dane, muszą one być dobrze sformułowane przy użyciu niektórych technologii semantycznych. Wiele semantycznych technologii opartych na języku XML może być używanych do wydobywania wiedzy, odkrywania i wykorzystywania zasobów oraz modelowania. Efektywna wymiana XML (EXI) jest często używana do rozwiązywania problemów z wydajnością, które wynikają z używania XML (np. parsowania XML), aby był odpowiedni dla aplikacji IoT. Wiele węzłów różnych marek może generować dane, które mają różną formę w każdym typie węzłów. Taki problem wymaga więcej pracy, aby ujednolicić dane tak, aby można je było efektywnie przetwarzać. Dlatego dynamiczna semantyka jest ważnym czynnikiem poprawy interoperacyjności IoT. Technologie IoT dla semantyki obejmują:

* Język modelu czujnika (SensorML). Standardowy model oparty na schemacie XML do opisu czujników i procedur pomiarowych. SensorML jest przydatny w systemach IoT do tworzenia elektronicznych arkuszy opisu modułów czujników i zbierania metadanych, które mają być wykorzystywane do wykrywania systemów czujników i obserwowania procesów. Umożliwia również autonomiczne sieci czujników dzięki samoopisującym się cechom czujników obsługujących SensorML.

* Typy nośników dla języka znaczników czujnika (SenML). Jest to nowy, prosty model pozyskiwania wykrytych danych i sterowania elementami wykonawczymi. Zapewnia semantykę danych i pozwala na dodatkowe metadane z linkami i rozszerzeniami. Ten prosty model może być używany w wielu aplikacjach IoT (Keränen i Jennings, 2017). Na przykład czujnik, taki jak czujnik wilgotności, może wykorzystywać ten rodzaj nośnika w CoAP do przesyłania pomiarów czujników.

* Baza danych IoT (IOTDB). Nowa technologia z nieograniczoną możliwością rozbudowy, która obsługuje semantykę w celu zapewnienia formalnych definicji wszystkich niezbędnych elementów. W przeciwieństwie do wyżej wymienionych technologii, IOTDB używa słowników JSON do manipulowania i monitorowania węzłów, co czyni ją stosunkowo szybką, ponieważ parsowanie JSON jest zawsze bardziej wydajne niż parsowanie XML. IOTDB jest kompatybilny z protokołami, takimi jak CoAP i MQTT.

* RESTful API Modeling Language (RAML). Język ten jest używany do definiowania interfejsów API opartych na HTTP, które reprezentują większość zasad Representational State Transfer (REST)60. Ponieważ jest zgodny z REST, jest bardziej prawdopodobne, że będzie używany w scenariuszach IoT, które są odpowiednie dla CoAP , gdzie obciążenie sieci jest znikome.

* Wolfram Data Drop. Otwarta usługa, która umożliwia gromadzenie danych dowolnego typu z dowolnego miejsca (w tym z węzłów IoT) w celu semantycznego przygotowania ich do natychmiastowych obliczeń, zapytań, analiz, wizualizacji lub innych operacji. Dane obliczalne (tj. kolekcje i serie czasowe) są zapisywane w nazwanych pojemnikach danych w chmurze Wolfram i są natychmiast dostępne ze wszystkich innych systemów/aplikacji.

Analiza Big Data

Środowiska IoT obejmują ogromną liczbę czujników, które gromadzą ogromne ilości danych, co skutkuje niezwykle dużymi zbiorami danych (tj. Big Data), które można analizować obliczeniowo w celu wyodrębnienia wiedzy, takiej jak wzorce, trendy i powiązania. Wygenerowane Big Data wciąż rośnie, ponieważ IoT gromadzi dodatkowe dane. Dlatego wymagane są specjalne technologie, które mogą obsłużyć coraz większą ilość danych. Najbardziej wydajną strategią adresowania stale rosnących danych jest przetwarzanie ich w czasie rzeczywistym (streaming). Przetwarzanie w czasie rzeczywistym umożliwia dostęp do aktualnych analiz, które odzwierciedlają ostatnie zmiany w danych i oszczędzają pamięć masową. Uczenie maszynowe może również wykorzystywać Big Data do tworzenia dokładnych prognoz. Dodatkowo, przetwarzanie równoległe może być stosowane przy użyciu niektórych węzłów IoT (lub jednostek obliczeniowych mgły) do przetwarzania danych równolegle z serwerami zaplecza w celu uzyskania lepszej wydajności i równoważenia obciążenia. Następujące technologie pomagają w generowaniu analityki w czasie rzeczywistym (streaming) z Big Data:

* Iskra Apache. Dobrze znana technologia przetwarzania rozproszonego typu open source, która wykorzystuje buforowanie w pamięci i oferuje ulepszone wykonanie w celu uzyskania lepszej wydajności. Apache Spark obsługuje analizy strumieniowe, przetwarzanie wsadowe, bazy danych wykresów, zapytania ad hoc i uczenie maszynowe. Chociaż Spark był używany w wielu systemach IoT, wymaga dodatkowego wysiłku, aby przezwyciężyć pewne ograniczenia, takie jak opóźnienia czasowe w przetwarzaniu w czasie rzeczywistym. Spark to bardzo elastyczna technologia, ponieważ obsługuje dane statyczne i strumieniowe oraz wiele języków programowania.

* Apache Apex. Platforma typu open source, podobna do platformy Spark, ujednolica przetwarzanie strumieniowe i wsadowe w sposób rozproszony, skalowalny, odporny na błędy, wydajny, stanowy i bezpieczny. W przeciwieństwie do Sparka ma bardzo małe opóźnienie i przetwarza bigdata w ruchu (tj. przetwarza dane strumieniowe, w tym modyfikację podczas lotu). Jego zdolność do przetwarzania

dużych zbiorów danych w ruchu umożliwia łatwą integrację z jednostkami obliczeniowymi mgły. Brakuje jednak przetwarzania poza kolejnością i

obsługuje tylko Javę jako język programowania. Chociaż jest to stosunkowo nowa platforma, jest szeroko stosowana w wielu projektach IoT na dużą skalę, takich jak inteligentne sieci i zastosowania przemysłowe.

* Apache Swift. Platforma open-source, która jest koncepcyjnie bardzo podobna do Apex. Jest jednak starszy niż Apex i nie był często używany w rzeczywistych projektach IoT ze względu na brak autoskalowania i ograniczone gwarancje dostarczania wiadomości (QoS).

* Apache Kafka. Biblioteka kliencka służąca do przetwarzania i analizowania danych przechowywanych w Kafce oraz przekazywania danych wyjściowych do systemu zewnętrznego lub zapisywania ich z powrotem do Kafki. Pierwszy scenariusz pozwala Kafce służyć jako warstwa środkowa, w której zbiera dane generowane przez czujniki IoT, następnie wykonuje na nich pewne operacje (np. filtrowanie i sortowanie), a na końcu przesyła je do innych komponentów w innych frameworkach (np. Flink i Spark).

Platformy IoT i systemy operacyjne

Istnieje ogromna liczba platform IoT i systemów operacyjnych, które mogą integrować wiele z wyżej wymienionych technologii w celu świadczenia usług IoT. W kontekście IoT oba terminy (tj. platformy i systemy operacyjne) są używane zamiennie. Jest jednak między nimi niewielka różnica, tzn. systemy operacyjne (tj. wbudowane systemy operacyjne lub RTOS) skupiają się tylko na niektórych funkcjach związanych z komunikacją, takich jak podłączanie czujników do

Internet i umożliwienie zbierania danych z czujników. Inne funkcje, takie jak analityka, często występują w razie potrzeby w systemach partnerskich. To sprawia, że systemy operacyjne są odpowiednie dla małych aplikacji IoT. Z drugiej strony platformy IoT obejmują prawie wszystkie funkcjonalności z pięciu wspomnianych warstw (np. komunikacja, wymiana danych, analityka itp.). Kolejną zaletą platform IoT jest łatwość konfiguracji i wdrożenia. Platformy IoT, w przeciwieństwie do systemów operacyjnych, które wymagają dodatkowego wysiłku, aby zintegrować się z innymi systemami, umożliwiają użytkownikom IoT łatwy wybór odpowiednich technologii w oparciu o ich potrzeby oraz zapewniają lepszą kompatybilność i wsparcie. W tej sekcji omówiono tylko popularne ogólne systemy operacyjne i platformy IoT. Poniżej przedstawiono dwa dobrze znane systemy operacyjne IoT oparte na języku C:

* RIOT. Lekki system operacyjny do bezprzewodowych systemów sieciowych z ograniczoną pamięcią, skupiający się na urządzeniach IoT o niskim poborze mocy. RIOT obsługuje szeroką gamę urządzeń i oferuje większą złożoność, która pozwala na większe i wrażliwe na czas aplikacje IoT.

Contiki. System operacyjny typu open source do łączenia z Internetem ograniczonych pamięci i energooszczędnych systemów bezprzewodowych. Jest podobny do RIOT, ale mniej zaawansowany pod względem opóźnień i wydajności (tj. wielowątkowości). Jednak jedną z głównych zalet Contiki jest jej symulator (czyli Cooja). Cooja pomaga naukowcom i programistom testować ich systemy przed zakupem urządzeń. Ponieważ Contiki jest stosunkowo starszy niż RIOT, jest szeroko stosowany, a zatem ma aktywną społeczność.

Można również używać innych systemów operacyjnych, takich jak Linux, ale ze względu na wymagania dotyczące większej pamięci ROM i RAM (tj. ∼1 MB), nie jest on odpowiedni dla małych urządzeń IoT. Tabela 3.4 zawiera porównanie głównych funkcji systemów operacyjnych RIOT i Contiki. RIOT OS jest bardziej zaawansowany, ponieważ obsługuje wielowątkowość, a także bardzo małe opóźnienia (prawie w czasie rzeczywistym). Większość platform IoT jest oparta na chmurze i zapewnia technologie IoT dla

większości funkcji wymienionych w poprzedniej sekcji. Oto niektóre platformy spośród ogromnej liczby obecnych platform:

*AWS IoT. Komercyjna platforma IoT, która wykorzystuje popularne usługi chmurowe AWS, aby zapewnić wszystko, czego potrzebuje przedsiębiorstwo (tj. zarządzanie urządzeniami, wizualizacja). Pozyskiwaniem, przechowywaniem, przetwarzaniem i wizualizacją danych zajmują się takie usługi, jak Amazon Redshift, Amazon EMR, AWS Lambda, Amazon DynamoDB, Amazon Kinesis i Amazon QuickSight.

IBM Watson. Komercyjna platforma IoT, która jest silnie zintegrowana z Bluemix, aby wnieść moc obliczeniowej kognitywnej i uczenia maszynowego do IoT. Platforma ta może być wdrożona w chmurze lub lokalnie, w której wprowadzanie urządzeń na platformę jest zautomatyzowane za pomocą zestawów SDK i interfejsów API. Pozwala również na blockchain, w którym integracja IoT z rozwijającą się technologią rozproszonych rejestrów opiera się na HyperLedger.

* ThingWorx. Komercyjna platforma do rozwoju inteligentnych urządzeń podłączonych do sieci za pomocą zintegrowanego narzędzia programistycznego IoT, które obsługuje łączność, produkcję, analizę i inne obszary IoT. ThingWorx pozwala użytkownikom łączyć urządzenia, ustanawiać źródło danych, ustalać zachowania urządzeń i budować interfejs bez żadnego kodowania.

Pakiet IoT firmy Bosch. Komercyjny, elastyczny IoT oparty na chmurze, który umożliwia programistom testowanie aplikacji przed ich wdrożeniem, wdrożeniem i eksploatacją w normalnych warunkach. Jego możliwości zarządzania urządzeniami (tj. wykonywanie procesów wdrażania oprogramowania, łączenie systemów i usług innych firm oraz analizowanie danych) mogą być również używane samodzielnie i lokalnie. Xively. Komercyjna platforma oparta na chmurze, która ułatwia opracowywanie połączonych inteligentnych produktów dla biznesu dzięki silnej współpracy z partnerami sprzętowymi i integracji za pomocą jednego kliknięcia z narzędziami biznesowymi. Xively umożliwia wizualizację danych graficznie w czasie rzeczywistym oraz zdalną aktualizację urządzeń.

* EVRYTHNG. Oparta na chmurze platforma do zarządzania tożsamościami IoT dla produktów, w której wszystkie produkty mają stałą, adresowalną obecność w Internecie. Umożliwia elastyczne semantyczne magazynowanie danych dostosowywanie tych dynamicznych profili danych dla dowolnego produktu, tak aby autoryzowane aplikacje mogły z nimi wchodzić w interakcje i wymieniać dane w czasie rzeczywistym podczas ich cyklu życia.

* Kaa. Oparta na chmurze platforma typu open source do zarządzania urządzeniami IoT i analizowania generowanych danych w celu zapewnienia kompletnych, kompleksowych rozwiązań IoT, połączonych aplikacji i inteligentnych produktów. Jest kompatybilny z praktycznie wszystkimi podłączonymi urządzeniami i bramami. Umożliwia używanie urządzeń prawie jako jednostek typu plug and play z minimalnym kodem.

Wniosek

W tej części w logiczny sposób, w oparciu o pięciowarstwowy model IoT, zaprezentowano aktualne i powszechnie stosowane technologie i protokoły oraz ich funkcjonalności. Omawiane technologie warstwy percepcji to ogólne urządzenia fizyczne, które można wykorzystać do wysyłania wyczuwanych danych. Technologie sieciowe wymienione w drugiej warstwie umożliwiają identyfikację węzłów i dotarcie do nich za pomocą bezpiecznego medium przy użyciu specjalnych protokołów routingu, takich jak RPL. Technologie omówione w warstwie oprogramowania pośredniczącego wykorzystują komponenty fizyczne do świadczenia usług, które są łatwo wykrywalne i zorganizowane do użytku przez użytkowników IoT w wyższych warstwach. Warstwa aplikacji zapewnia ostateczny cel systemów

IoT (tj. usług), aby umożliwić użytkownikom końcowym dostęp do nich i korzystanie z nich na ich ulubionej platformie. Wreszcie, technologie w warstwie biznesowej pomagają administratorom systemów monitorować i ulepszać działania IoT poprzez analizy wykonywane na zebranych danych. Technologie zostały omówione w sposób ogólny, bez sugerowania jakichkolwiek preferencji, ponieważ nie można polecić najbardziej odpowiednich technologii IoT bez znajomości wymagań konkretnego scenariusza, w którym zostaną zastosowane. Na przykład IoMT prawdopodobnie wymaga niskich opóźnień i bezpiecznych technologii; Dlatego wybór technologii, które mają takie cechy, byłby właściwy. Porównania informacyjne w tej części uwypukliły główne różnice między różnymi technologiami. Z obecnie dostępnymi technologiami IoT pozostaje kilka wyzwań. Po pierwsze, w każdej warstwie potrzebna jest lepsza integracja pozioma między protokołami, aby umożliwić istniejącym aplikacjom IoT współpracę i współdzielenie zasobów. Na przykład w warstwie oprogramowania pośredniczącego system CoAP powinien być w stanie zintegrować się z DDS. Ta integracja umożliwi współpracę między organizacjami i pozwoli im efektywnie dzielić się zasobami (np. czujnikami) lub danymi. Po drugie, ciągłość usług mobilnego Internetu pozostaje wyzwaniem, ponieważ wszystkie obecne technologie opierają się na stałych punktach (np. wieżach telefonii komórkowej i routerach granicznych) w celu zapewnienia łączności. Po trzecie, skalowalność i interoperacyjność Internetu Rzeczy pozostaje wyzwaniem, ponieważ problemy ze zgodnością są nieuniknione w obecnym stanie, ponieważ w grę wchodzi wiele różnych technologii. Wreszcie, bezpieczeństwo w IoT to wyraźny problem, który dotyka obecnie dostępne technologie ze względu na potencjalny wpływ udanych ataków na dużą skalę. Technologie wspierające IoT są kluczem do tworzenia skutecznych i skutecznych aplikacji IoT. Ponieważ liczba technologii stale rośnie, zrozumienie cech i ograniczeń obecnie dostępnych technologii jest ważne, ponieważ stanowią podstawę dla nowych technologii. Mamy nadzieję, że obecne wyzwania w IoT zostaną zminimalizowane przez szybki rozwój technologii IoT napędzanych przez wiele organizacji na całym świecie.