

Z1. Monitorowanie usług w sieciach IP

Uwaga: warto rozważyć, czy realizowane zadanie laboratoryjne da się wykorzystać do rozszerzenia własnej pracy dyplomowej o część praktyczną lub badawczą. To zadanie może okazać się szczególnie przydatne dla studentów realizujących prace dyplomowe związane z konfiguracją i badaniem pracy serwerów różnych usług sieciowych.

Wprowadzenie

Cyklicznie monitorowanie usług w sieciach IP jest czynnością, którą muszą wykonywać podmioty odpowiedzialne za utrzymanie nieprzerwanej pracy serwerów świadczących te usługi. Olbrzymia ilość usług marketingowych, handlowych, finansowych i in., realizowanych poprzez sieć Internet powoduje, że wiele firm zdecydowało się zrezygnować z samodzielnego utrzymania, nadzorowania, serwisowania serwerów świadczących określone usługi i powierzyć opiekę nad nimi firmom outsourcingowym. Firmy te są często bardzo dużymi, wyspecjalizowanymi korporacjami, zatrudniającymi specjalistów monitorujących pracę serwerów, a także świadczących usługi serwisowe, związane m.in. z utrzymaniem, optymalizacją i przywróceniem systemów i usług po awariach. Firmy takie posiadają wielojęzyczną kadrę, która obsługuje BOK (Biuro Obsługi Klienta / Call Center), przyjmujące w systemie 24/7 zgłoszenia awarii od klientów. Utrzymanie w nieprzerwanej gotowości wielojęzycznej kadry, w celu przyjmowania i obsługi zgłoszeń awarii jest bardzo kosztowne. O wiele lepszym rozwiązaniem jest tworzenie oprogramowania do automatycznego monitorowania wybranych usług, uruchomionych na serwerach i grupach serwerów obsługiwanych przez daną korporację.

Specyfikacja zadania

1. Zadanie ma na celu stworzenie załączka aplikacji, która będzie realizowała m.in. niektóre z wymienionych poniżej funkcji:
 - gromadzenie informacji o zasobach klientów, którymi należy się opiekować:
 - nazwy domenowe/numery IP serwerów klientów,
 - porty, na których działają poszczególne usługi na poszczególnych serwerach,
 - częstotliwości z jakimi należy monitorować wybrane usługi (tzn. porty na określonych serwerach),
 - ilości powtórzeń prób nawiązania połączenia z poszczególnymi usługami lub inne kryteria, na podstawie których należy uznać, że poszczególne usługi nie działają prawidłowo i należy zająć się ich przywróceniem,
 - gromadzenie informacji o klientach:
 - nazwa, adres klienta,
 - dane kontaktowe osób zajmujących się eksploatacją usług, których utrzymanie zlecono,
 - gromadzenie informacji o pracownikach własnej firmy, zobowiązanych do utrzymania usług klientów:
 - listy klientów/serwerów/usług, którymi się opiekują poszczególni pracownicy,
 - specjalność, pracownika, jego certyfikaty i doświadczenie zawodowe, języki, w których można się z nimi porozumieć,
 - realizacja monitorowania działania poszczególnych usług na poszczególnych serwerach i obsługa wykrytych awarii:
 - gromadzenie historii dotyczącej wykrytych awarii usług,
 - gromadzenie historii dotyczącej wygenerowanych monitów, przekazanych do pracowników odpowiedzialnych za obsługę awarii poszczególnych usług na poszczególnych komputerach,
 - gromadzenie historii potwierdzeń przyjęcia monitów przez pracowników odpowiedzialnych za obsługę awarii poszczególnych usług na poszczególnych komputerach,
 - gromadzenie historii dotyczącej klasyfikacji występujących awarii usług, wyspecyfikowanych przez obsługujących je pracowników oraz zastosowanych przez nich działań zaradczych (szczególnie ten ostatni aspekt jest istotny, ponieważ pozwala na zbudowanie bazy danych sprawdzonych rozwiązań, które mogą być wskazówką przyszłych działań, szczególnie podejmowanych przez dopiero wdrażających się pracowników II linii wsparcia),
 - generowanie i gromadzenie danych statystycznych związanych z wyznaczaniem i optymalizacją kosztów obsługi poszczególnych klientów, w związku z obsługą przypisanych do nich zadań, realizowanych np. na podstawie ilości generowanych zadań, wynikających z awarii usług oraz czasu ich obsługi.
2. Wymienione powyżej funkcje stanowią zaledwie kilkanaście procent specjalistycznego systemu CRM (Customer Relationship Management - Zarządzanie relacjami z klientami) do obsługi klientów zlecających utrzymanie usług sieciowych, a i tak ich pełna realizacja przekroczyłaby czas przewidziany na zajęcia laboratoryjne. Dlatego poniżej przedstawiono minimalną listę wymaganych czynności niezbędnych, aby stworzyć draft aplikacji sieciowej, do prezentacji jej funkcjonalności dla klienta lub project managera, który mógłby zdecydować o jej dalszym rozwoju lub kształcie.

Realizacja zadania

1. Wszystkie czynności realizuj z wykorzystaniem własnego hostingu, ewentualnie współdzielonego hostingu, jeśli nie posiadasz własnego hostingu.
2. Definiuj w każdym webowym dokumencie stronę kodową UTF-8, skonfiguruj też odpowiednio swoje narzędzia i bazę danych.
3. Głównemu dokumentowi tworzonemu na swoim hostingu nadawaj tytuł taki, jak Twoje nazwisko, to bardzo ułatwia Twojemu belfrowi współbieżne sprawdzanie prac wielu studentów. Z góry dziękuję za stosowanie się do tej prośby ☺.

```
<head>
  <title>Twoje nazwisko</title>
</head>
```

4. Przetestuj na swoim hostingu działanie różnych skryptów PHP do monitorowania działania usług na wybranych portach, uruchomionych na wybranych hostach. Pamiętaj, że nie należy do testowania swoich skryptów używać hostów rządowych (.gov), ani innych hostów należących do instytucji rządowych i państwowych, ponieważ skanowanie portów może być traktowane przez ich administratorów, jako wstęp do nielegalnych działań hackerskich. Bez obawy o posądzenie o łamanie prawa można do takich prób używać adresów popularnych portali informacyjnych lub społecznościowych np. fb.com itd. Przeczytaj opis polecenia fsockopen <http://php.net/manual/en/function.fsockopen.php>

```
<?php
    $host = 'fb.com';
    $port = '80';
    {
        $fp = @fsockopen($host, $port, $errno, $errstr, 30);
        echo 'Host '.$host.':'.$port.' jest: ';
        if ($fp) { echo 'OK'; } else { echo 'Down'; }
        if (!$fp) { echo "$errstr ($errno)"; }
    }
?>
```

5. Za pomocą phpMyAdmina stwórz bazę danych. W tej bazie stwórz tabelę o nazwie **domeny**. W tabeli tej umieść dwie kolumny
- id – typu int (autonumerowanie), klucz główny tabeli,
 - adres – typu tekstowego.
6. Wstaw do tej tabeli 5 nazw hostów, które będą później wykorzystane do cyklicznego monitorowania np.

+ Opcje

				id	adres
<input type="checkbox"/>	Edytuj	Kopiuj	Usuń	1	wp.pl
<input type="checkbox"/>	Edytuj	Kopiuj	Usuń	2	utp.edu.pl
<input type="checkbox"/>	Edytuj	Kopiuj	Usuń	3	interia.pl
<input type="checkbox"/>	Edytuj	Kopiuj	Usuń	4	youtube.com
<input type="checkbox"/>	Edytuj	Kopiuj	Usuń	5	fb.com

7. Napisz skrypt w PHP generujący na Twojej stronie tabelę ze znaczników HTML, zawierającą nazwy wprowadzonych hostów

```
<body>
<?php
    $dbhost="Twój_host"; $dbuser="Twoje_konto"; $dbpassword="Twoje_hasło"; $dbname="Twoja_BD";
    $polaczenie = mysqli_connect ($dbhost, $dbuser, $dbpassword);
    mysqli_select_db ($polaczenie, $dbname);
    $rezultat = mysqli_query ($polaczenie, "SELECT * FROM domeny");
    print "<TABLE CELLPADDING=5 BORDER=1>";
    print "<TR><TD>idt</TD><TD>Nazwa</TD></TR>\n";
    while ($wiersz = mysqli_fetch_array ($rezultat))
    {
        $idt = $wiersz [0];
        $nazwa = $wiersz [1];
        print "<TR><TD>$idt</TD><TD>$nazwa</TD></TR>\n";
    }
    print "</TABLE>";
?>
</body>
```

8. Rezultat wykonania tego skryptu powinien wyglądać tak, jak poniżej, obok pokazano wygląd tabeli w następnym kroku.

idt	Nazwa
1	wp.pl
2	utp.edu.pl
3	interia.pl
4	youtube.com
5	fb.com

idt	Nazwa	Pracuje
1	wp.pl	OK
2	utp.edu.pl	OK
3	interia.pl	OK
4	youtube.com	OK
5	fb.com	OK



9. Zmodyfikuj skrypt w taki sposób, aby wyświetlał nazwy hostów oraz informacje czy port 80 tych hostów jest otwarty – "OK".

10. Za pomocą phpMyAdmina wprowadź do tabeli **domeny** kolejną nazwę hosta, ale tym razem nieistniejącą np. wp.ppl, aby się przekonać, czy skrypt badania hostów działa prawidłowo.
11. Aby móc na bieżąco obserwować wyniki działania swojego skryptu, czyli poprawną pracę hostów zdefiniowanych w BD, musisz cyklicznie odświeżać stronę (w większości przeglądarek - klawisz F5). Na dłuższą metę jest to niewygodne, dlatego dodaj do swojego skryptu automatyczne przeładowywanie co 10 sekund:

```
<head>
  <meta http-equiv="refresh" content="10" />
</head>
```

12. Rozbuduj swoją aplikację o niektóre elementy wymienione poniżej, które zostaną dla Twojej grupy określone przez prowadzącego zajęcia
- Rozbuduj tabelę o kolejną kolumnę, w której wyświetlaj ilość prób komunikacji z danym hostem, które pozostały bez odpowiedzi. Jeśli host zaczyna ponownie odpowiadać – to ponownie wyświetlaj w tej kolumnie cyfrę "0".
 - Rozbuduj tabelę o kolejną kolumnę, w której wyświetlaj czas, od którego nastąpiła utrata komunikacji z danym hostem. Jeśli host zaczyna ponownie odpowiadać – to ponownie wyświetlaj w tej kolumnie znak "-".
 - Rozbuduj tabelę o kolejną kolumnę, aby wyświetlać łączny czas, przez który host nie był dostępny od początku czasu, w którym go monitorujesz. W związku z tym, że skrypt jest wywoływany cyklicznie co 10s – łatwo przeliczyć ilość prób komunikacji, które pozostały bez odpowiedzi na czas, w którym nie było komunikacji z danym hostem. Czas ten wyraż w sekundach i minutach.
 - Utwórz subdomenę własnej domeny i postaw na niej prostą stronę WWW, tworząc w jej katalogu macierzystym plik index.html. Sprawdź, czy jest ona dostępna. Wprowadź nazwę tej subdomeny na listę monitorowanych hostów. Usuń i ponownie wstawiaj plik do katalogu macierzystego subdomeny i jednocześnie sprawdź, jak się to przekłada na statystykę monitorowanego hosta.
 - Rozbuduj swoją aplikację w ten sposób, aby było możliwe tworzenie formularza, za pomocą którego dodaje się nowy adres serwera do monitorowania.
13. Zgłoś realizację zadania.
14. Pełne zakończenie zadania polega na przesłaniu sprawozdania zrealizowanego zadania na mail prowadzącego zajęcia. Sprawozdanie powinno składać się z następujących elementów:
- Zbiór wszystkich plików tworzących kod źródłowy portalu, skompresowanych do archiwum, z uwzględnieniem struktury katalogów. Pliki źródłowe aplikacji powinny zawierać podstawowe komentarze stworzonego kodu. Ze względów bezpieczeństwa można zastępować w sprawozdaniu hasło do bazy danych krzyżykami XXXXX.
 - Plik generujący strukturę bazy danych.
 - Plik z dokumentacją projektu w formacie Word lub PDF. Powinien on zawierać na pierwszej stronie standardową tabelkę z nazwą uczelni, kierunkiem studiów, nazwą przedmiotu, nazwiskiem studenta, nr indeksu studenta, nazwiskiem opiekuna przedmiotu oraz numerem i nazwą zadania laboratoryjnego (np. Z1 Monitorowanie usług w sieciach IP). Wzór takiej tabelki podano poniżej. Na drugiej stronie powinien znajdować się krótki opis stworzonej aplikacji i realizowanych przez nią funkcjonalności. Na dalszych stronach należy umieścić zrzuty ekranowe przedstawiające podstawowe funkcje programu i najciekawsze fragmenty realizującego je kodu, opatrzone stosownymi komentarzami. Zwykle objętość dokumentacji nie powinna przekraczać 7 stron.
 - UWAGA:** osoby, które zdążą zrobić zadanie w czasie zajęć laboratoryjnych, na których jego treść została przekazana - nie muszą dołączać do sprawozdania pliku z dokumentacją projektu, wysyłają jedynie na mail prowadzącego pliki źródłowe swojej aplikacji oraz plik generujący strukturę bazy danych, a zamiast sprawozdania załączają pusty plik tekstowy o nazwie ZALICZONE NA ZAJĘCIACH.TXT.

Wzór tabelki do sprawozdania

 <div> Uniwersytet Technologiczno-Przyrodniczy im. J.J. Śniadeckich w Bydgoszczy Wydział Telekomunikacji, Informatyki i Elektrotechniki Zakład Techniki Cyfrowej </div> 			
Przedmiot:	Programowanie aplikacji sieciowych		Teleinformatyka, Studia stacjonarne semestr 7, 2017/2018
Temat:	Monitorowanie usług w sieciach IP (proszę pamiętać, aby to zmieniać ☺)		
Numer lab.:	1	Data wykonania:	RRRR.MM.DD
Prowadzący:	dr. inż. P. Grad	Data oddania:	RRRR.MM.DD
Autor:	Imię i Nazwisko	Indeks:	XXXXXX