

Linux Bash Scripting / SystemD Customization

Einführung

Ein großer Vorteil von Linux: Customization

- Linux bietet umfassende Anpassungsmöglichkeiten für Nutzer.
- Durch die Nutzung von Skripten und Systemdiensten lässt sich das Verhalten des Systems erheblich erweitern und personalisieren.

Szenario

- Ziel ist es, durch die Entwicklung einer Linux Applikation/Erweiterung, wichtige Linux Konzepte zu vertiefen.
- Die Applikation soll in der Terminal-Prompt (ubuntu@hostname(~)) Informationen zum Systemstatus anzeigen, wie z.B. die Latenz eines Pings.
- Dieses Projekt demonstriert die Flexibilität von Linux, um Anpassungen in verschiedenen Systembereichen vorzunehmen.

Kompetenzen

- Um das Projektziel zu erreichen, werden folgende Fähigkeiten benötigt:
 - ◆ Bash Scripting: Erstellen und Ausführen von Shell-Skripten.
 - ◆ Erstellen einer SystemD Unit: Automatisierung von Skripten beim Systemstart oder in regelmäßigen Intervallen.
 - ◆ Änderung der .bashrc Datei: Anpassen der Terminal-Prompt.

Konzepte

BashRc

- Die .bashrc Datei ist ein Skript, das jedes Mal ausgeführt wird, wenn eine neue Bash-Shell gestartet wird. Dies gilt insbesondere für interaktive, nicht-login Shells.
- In der .bashrc können verschiedene Einstellungen und Umgebungen für die Shell-Sitzung definiert werden. Dazu gehören:
 - ◆ Aliasdefinitionen: Kurzbefehle für längere Befehlssequenzen.
 - ◆ Funktionsdeklarationen: Benutzerdefinierte Funktionen, die bestimmte Aufgaben ausführen.
 - ◆ Umgebungsvariablen: Festlegen oder Modifizieren der Variablen, die das Verhalten der Shell beeinflussen.
 - ◆ Shell-Optionen: Anpassen des Verhaltens der Shell, z.B. durch Aktivieren der Autovervollständigung.
 - ◆ Prompt-Anpassung: Ändern des Aussehens der Befehlszeile (Prompt), z.B. durch Hinzufügen von Farben, Informationen über das aktuelle Verzeichnis oder Systeminformationen.
- Die .bashrc Datei ermöglicht es Benutzern, ihre Shell-Umgebung an ihre Bedürfnisse und Vorlieben anzupassen.

Prompt

- Der Prompt in der Linux-Shell ist die Textzeile, die anzeigt, dass die Shell bereit für die Eingabe eines Befehls ist.
- Standardmäßig zeigt der Prompt Informationen wie den Benutzernamen, den Hostnamen und das aktuelle Verzeichnis an.
- Durch Anpassen des Prompts in der `.bashrc`-Datei können zusätzliche Informationen oder Formatierungen hinzugefügt werden:
 - ◆ Farben und Formatierung: Hervorheben von Teilen des Prompts mit verschiedenen Farben oder Schriftarten.
 - ◆ Informationen hinzufügen: Anzeigen von Systeminformationen wie Uhrzeit, Datum, Last, Anzahl der laufenden Prozesse, Git-Branch-Informationen usw.
 - ◆ Dynamische Inhalte: Einbinden von Skript-Ergebnissen oder Befehlsausgaben, wie z.B. die Anzeige der aktuellen Latenz, wie in unserem Szenario.
- Die Anpassung des Prompts verbessert die Benutzererfahrung und bietet einen schnellen Überblick über wichtige Informationen.

SystemD - Init System

- SystemD ist ein System- und Dienstmanager, der in vielen modernen Linux-Distributionen verwendet wird.
- Es ersetzt traditionelle Init-Systeme wie SysVinit und Upstart und bietet effizientere Möglichkeiten, um Dienste und Prozesse zu verwalten.
- Hauptmerkmale von SystemD:
 - ◆ Unit-Dateien: Konfigurationsdateien, die Dienste, Mountpoints, Geräte und andere Ressourcen beschreiben, die SystemD verwalten kann.
 - ◆ Parallele Ausführung: Beschleunigt den Boot-Prozess durch paralleles Starten von Diensten.
 - ◆ Abhängigkeitsmanagement: Automatisches Starten und Stoppen von Diensten basierend auf ihren Abhängigkeiten.
 - ◆ Logging und Journaling: Zentrales Logging-System, das die Überwachung und Fehlerbehebung erleichtert.
 - ◆ Dynamische Konfiguration: Dienste können zur Laufzeit hinzugefügt oder geändert werden, ohne das gesamte System neu starten zu müssen.
- Durch die Verwendung von SystemD für die Automatisierung von Skripten und Diensten können Benutzer komplexe Aufgaben vereinfachen und die Systemleistung verbessern.

Aufbau der Applikation

Bash Script zur Ermittlung der Latenz

- Das Herzstück der Applikation ist ein Bash-Skript, dessen Hauptaufgabe es ist, die Netzwerklatenz zu einem vorgegebenen Server zu messen.
- Funktionsweise des Skripts:
 - ◆ Es führt periodisch den `ping`-Befehl aus, um die Latenzzeit (in Millisekunden) zu ermitteln.
 - ◆ Das Skript berechnet die durchschnittliche Latenz über einen bestimmten Zeitraum, um eine verlässliche Messung zu gewährleisten.

- ◆ Nach der Berechnung schreibt das Skript den durchschnittlichen Latenzwert in eine Textdatei. Diese Datei dient als Schnittstelle zwischen dem Skript und der Shell.
- Ziel: Durch die Speicherung der Latenzdaten in einer Datei wird vermieden, dass bei jedem neuen Shell-Start eine Latenzmessung durchgeführt werden muss, was die Leistung verbessern und Wartezeiten reduzieren kann.

SystemD Unit

- Eine SystemD Unit wird erstellt, um das Latenzmessung-Skript regelmäßig und automatisiert auszuführen.
- Konfiguration der Unit:
 - ◆ Die Unit wird so konfiguriert, dass sie das Bash-Skript in festgelegten Intervallen ausführt, beispielsweise alle 5 Minuten.
 - ◆ Dies stellt sicher, dass die in der Textdatei gespeicherten Latenzdaten aktuell bleiben, ohne eine manuelle Ausführung des Skripts zu erfordern.
- Vorteile: Die Verwendung von SystemD ermöglicht eine zuverlässige und effiziente Ausführung des Skripts im Hintergrund und entkoppelt die Latenzmessung von der direkten Interaktion des Benutzers mit der Shell.

BashRc -> Prompt

- Die Anpassung der .bashrc-Datei ist der letzte Schritt, um die Latenzinformation in der Terminal-Prompt anzuzeigen.
- Anpassungsprozess:
 - ◆ Ein kleiner Codeabschnitt wird der .bashrc hinzugefügt, der die Textdatei liest, in der das Bash-Skript die Latenzdaten speichert.
 - ◆ Dieser Code extrahiert den Latenzwert aus der Datei und integriert ihn in den Prompt.
- Effizienz: Da die .bashrc nur die bereits berechneten Daten aus der Textdatei liest, wird die Startzeit einer neuen Shell nicht beeinträchtigt. Die Latenzinformation wird sofort beim Öffnen einer neuen Terminal-Sitzung angezeigt, ohne zusätzliche Berechnungen oder Verzögerungen.
- Zusammenhang: Durch diese drei Komponenten - das Bash-Skript, die SystemD Unit und die .bashrc-Anpassung - entsteht eine nahtlose Integration der Latenzanzeige in den Prompt. Jeder Teil spielt eine spezifische Rolle: Das Skript ermittelt die Daten, SystemD sorgt für die regelmäßige Ausführung und die .bashrc präsentiert die Ergebnisse dem Benutzer auf effiziente Weise.