



SCRIPTING E AUTOMAÇÃO AVANÇADA

Responsável: João Pedro Toledo Gonçalves

Data: 26/01/2026

Código: ITGENG 0012/26 | **Classificação:** CONFIDENCIAL

Responsável: João Pedro Toledo Gonçalves | **Data:** 26/01/2026

1. HISTÓRICO DE REVISÃO

Data	Versão	Descrição	Autor
26/01/2026	1.0	Criação Inicial	João Pedro Toledo Gonçalves

2. OBJETIVO

Padronizar a criação de scripts Shell (Bash), agendamento de tarefas (Cron), automação de relatórios (AWK) e notificações por e-mail para otimizar a administração de servidores.

3. PRÉ-REQUISITOS

- ☐ Conhecimento de lógica de programação básica.
- ☐ Servidor SMTP (Exchange Interno ou O365) disponível para relay.

4. SHELL SCRIPTING BÁSICO

Todo script na iT Guys deve seguir este cabeçalho:

```
#!/bin/bash
# -----
# Script: backup_diario.sh
# Descricao: Realiza backup e envia logs.
# Autor: João Pedro Toledo Gonçalves
# Data: {{DATA_ATUAL}}
# -----

# Variaveis
DESTINO="/mnt/backup"
LOG="/var/log/meu_backup.log"

# Funcao de Log com Data
log_msg() {
    echo "$(date '+%Y-%m-%d %H:%M:%S') - $1" >> "$LOG"
}

# Execucao
log_msg "Inicio do Backup"
tar -czf "$DESTINO/backup.tar.gz" /etc/
if [ $? -eq 0 ]; then
    log_msg "Sucesso"
else
    log_msg "Erro no Backup"
fi
```

NOTA

NOTA ALPINE: O Alpine não traz o `bash` por padrão, apenas o `ash` (BusyBox).

- * **Opção A:** Mude a primeira linha para `#!/bin/sh` (Compatível).
- * **Opção B:** Instale o Bash: `apk add bash`.
- ? **DICA:** Execute com `bash -x script.sh` para debugar linha a linha.

5. AGENDAMENTO (CRON)

Use o agendador nativo para rodar scripts automaticamente.

1. [x] Edite a tabela do usuário: `crontab -e`
2. [x] Sintaxe: `m h dom mon dow command`
3. [x] Exemplos Úteis:

```
# Rodar todo dia as 03:00 da manha
00 03 * * * /home/scripts/backup_diario.sh

# Rodar a cada 1 hora
00 * * * * /usr/bin/php /var/www/html/cron.php

# Rodar todo dia 1 do mes
00 05 1 * * /home/scripts/relatorio_mensal.sh
```

6. NOTIFICAÇÕES POR E-MAIL (SMTP)

Para scripts enviarem alertas, configure um cliente SMTP simples como o `msmtp` ou `postfix` (relay).

Exemplo de Envio em Script (usando `mail` ou `sendmail`):

```
ASSUNTO="Alerta: Disco Cheio no Servidor $(hostname)"
EMAIL="monitoramento@itguys.com.br"
CORPO="O servidor esta com 95% de disco ocupado. Verifique."

echo "$CORPO" | mail -s "$ASSUNTO" -r "noreply@itguys.com.br" "$EMAIL"
```

Configuração SMTP (Exemplo genérico `/etc/postfix/main.cf`):

```
relayhost = [smtp.office365.com]:587
smtp_sasl_auth_enable = yes
smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd
smtp_tls_security_level = encrypt
```

7. RELATÓRIOS CUSTOMIZADOS (AWK)

O `awk` é perfeito para extrair colunas de logs.

Exemplo: Listar os 5 IPs que mais acessaram o Nginx

```
# Log format: IP - - [Data] "GET..."
awk '{print $1}' /var/log/nginx/access.log | sort | uniq -c | sort -nr | head -5
```

- `print \$1`: Pega a primeira coluna (IP).
- `sort | uniq -c`: Conta ocorrências.
- `head -5`: Pega o top 5.

8. MICRO APPS (ALIASES)

Crie atalhos para comandos longos no `~/bashrc`:

```
# Atualizar tudo com 1 palavra
alias updatedb='sudo apt update && sudo apt upgrade -y'

# Ver logs coloridos do nginx
alias logweb='tail -f /var/log/nginx/error.log | ccze'

# Limpar cache de memoria
alias dropcache='sync; echo 3 | sudo tee /proc/sys/vm/drop_caches'
```

Aplique com `source ~/bashrc`.

9. VALIDAÇÃO FINAL

- ☐ O script roda manualmente sem erros?
- ☐ O Cronjob apareceu no log (`grep CRON /var/log/syslog`)?
- ☐ O e-mail de teste chegou na caixa de correio?