

Dokumentation für das korrigierte NVMe-Kopierskript

Dieses Dokument beschreibt die Änderungen, die am ursprünglichen Skript `robust_nvme_copy.sh` vorgenommen wurden, um ein bootfähiges QEMU-Image einer Windows-Installation zu erstellen.

Probleme mit dem ursprünglichen Skript

Das ursprüngliche Skript hatte ein grundlegendes Problem, das verhinderte, dass das erstellte Image bootfähig war:

- **Kopieren einer einzelnen Partition:** Das Skript kopierte nur die Windows-Partition (`/dev/nvme1n1p3`) und nicht die gesamte NVMe-Festplatte. Für ein bootfähiges System, insbesondere mit Windows, ist es jedoch unerlässlich, die gesamte Festplatte zu kopieren. Dazu gehören der Master Boot Record (MBR) oder die GUID Partition Table (GPT) sowie alle zugehörigen Boot- und Wiederherstellungspartitionen.

Vorgenommene Korrekturen und Verbesserungen

Um dieses Problem zu beheben und das Skript zu verbessern, wurden folgende Änderungen vorgenommen:

1. **Kopieren der gesamten Festplatte:** Die Variable `SOURCE_DEVICE` wurde von `/dev/nvme1n1p3` auf `/dev/nvme1n1` geändert. Dadurch wird die gesamte NVMe-Festplatte kopiert, einschließlich aller Partitionen und der für den Start erforderlichen Boot-Informationen.
2. **Konvertierung in das qcow2-Format:** Das Skript konvertiert das erstellte RAW-Image nun automatisch in das `qcow2` -Format. Dieses Format bietet gegenüber dem RAW-Format mehrere Vorteile, darunter:
 - **Geringere Dateigröße:** `qcow2` unterstützt Thin Provisioning, was bedeutet, dass die Image-Datei nur so viel Speicherplatz belegt, wie tatsächlich Daten vorhanden sind.
 - **Snapshots:** Das `qcow2` -Format ermöglicht das Erstellen von Snapshots, mit denen Sie den Zustand Ihrer virtuellen Maschine zu einem bestimmten Zeitpunkt speichern und wiederherstellen können.
 - **Bessere Leistung:** In vielen Fällen bietet `qcow2` eine bessere Leistung als das RAW-Format.

3. **Aktualisierter QEMU-Befehl:** Der QEMU-Befehl zum Starten der virtuellen Maschine wurde aktualisiert, um das `qcow2` -Image zu verwenden und den UEFI-Modus zu aktivieren. Der neue Befehl lautet:
- `-drive file=${TARGET_FILE},format=qcow2,if=virtio` : Gibt an, dass das `qcow2` -Image mit dem `virtio` -Treiber für eine bessere Leistung verwendet werden soll.
 - `-bios /usr/share/ovmf/OVMF.fd` : Weist QEMU an, das OVMF-BIOS zu verwenden, das für den Start von UEFI-Systemen wie modernen Windows-Installationen erforderlich ist.

Verwendung des korrigierten Skripts

1. **Stellen Sie sicher, dass QEMU und `qemu-utils` installiert sind:**
2. **Führen Sie das korrigierte Skript mit `sudo` aus:**
3. **Folgen Sie den Anweisungen des Skripts**, um die externe SSD auszuwählen und den Kopiervorgang zu starten.
4. **Nach Abschluss des Vorgangs** finden Sie das bootfähige `qcow2` -Image im angegebenen Verzeichnis auf Ihrer externen SSD.
5. **Starten Sie die virtuelle Maschine** mit dem im Skript angegebenen QEMU-Befehl.