BAR - Burn After Reading

A secure, offline file management system with self-destruction capabilities and advanced file scanning.

R **by Rolan Lobo**

Overview

Completely Offline

Works without internet connection or server dependencies, ensuring your sensitive data never leaves your machine.
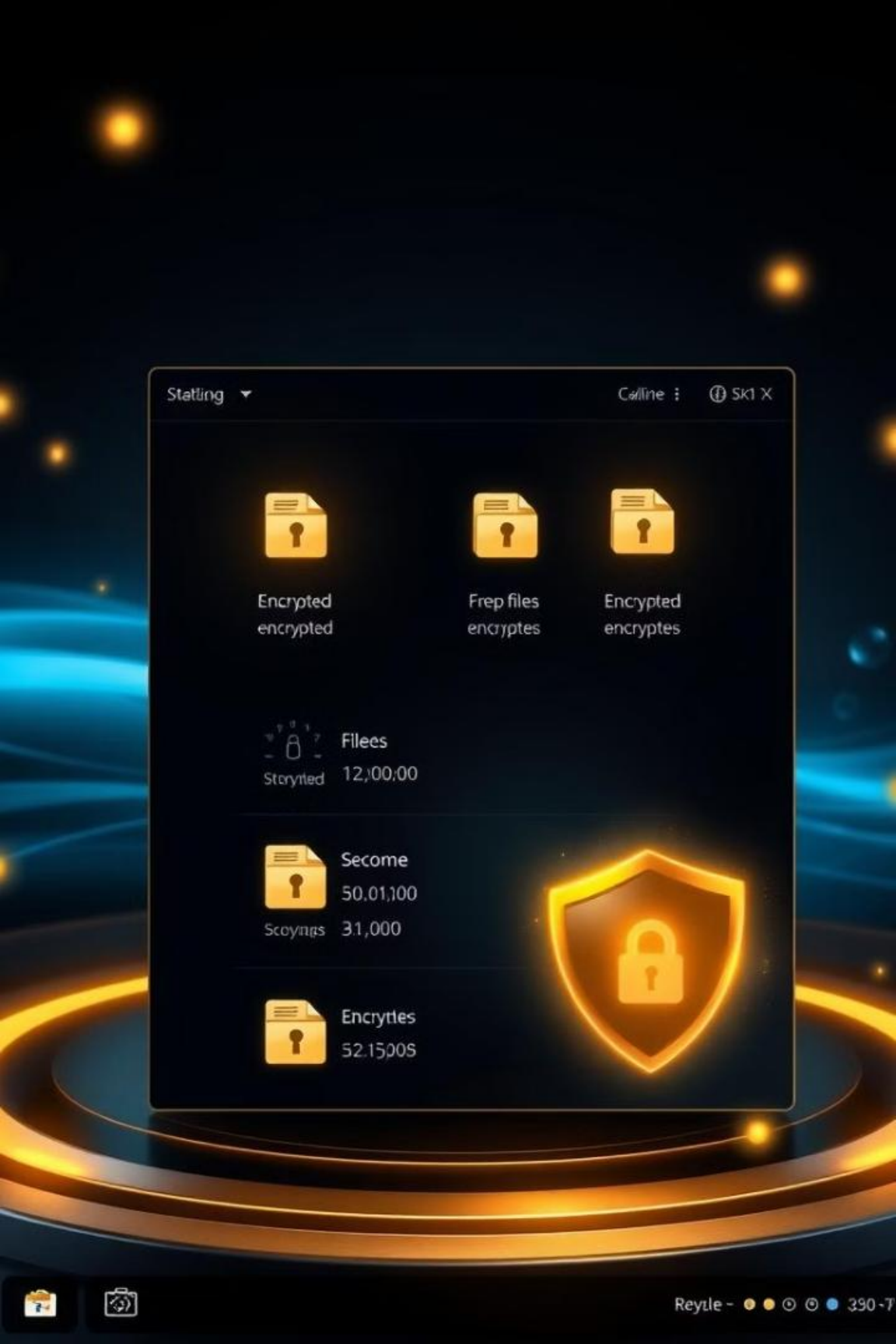
Strong Encryption

AES-256 encryption for all file content with secure key management.

Self-Destruction

Multiple mechanisms including time-based expiration, access count limits, and deadman switch functionality.

Advanced Scanning

Powerful file scanning capabilities to detect and manage .bar files across your devices.

Security Features

Encryption

- AES-256 in GCM mode for authenticated encryption
- Unique encryption key for each file

- Key derivation using PBKDF2-HMAC-SHA256

Self-Destruction

- Time-based: Files automatically delete after a specified time
- Access-count: Files delete after being accessed a certain number of times
- Deadman switch: Files delete if not accessed within a specified period

Secure Storage

- All data stored locally in the ~/.bar directory with proper encryption
- No plaintext storage of sensitive information
- Secure key management

File Scanning Capabilities

Signature Validation

Ensures file integrity through comprehensive validation checks

Version Compatibility

Checks for compatibility across different versions of BAR files

Multi-Device Detection

Secure detection of .bar files across all connected devices

Removable Media Support

Support for scanning external drives and removable storage

Multi-Threaded Performance

Scanning engine uses multiple threads for improved performance

Installation Options

Running the Executable

1. Download the latest release of BAR.exe

2. No installation required - simply double-click the executable to run

Running from Source

1. Ensure you have Python 3.8 or higher installed

2. Clone or download the repository

3. Install dependencies: pip install -r requirements.txt

4. Run the application: python main.py

Building Your Own Executable

1. Install dependencies: pip install -r requirements.txt

2. Run the build script: python build.py

3. Find the executable in the dist directory

Usage Guide

First-Time Setup

Create a local user account with a strong password

Login

Access your secure file storage

File Operations

- Create new secure files with custom security settings
- Access existing files (subject to security constraints)
- Export files for sharing
- Import shared files
- Scan devices for .bar files

Configuration

Adjust application settings and themes

Technical Details

Technology Stack

Built with Python 3.8+ and PyQt5

Packaging

Packaged as a standalone Windows executable using PyInstaller

Connectivity

No external dependencies or internet connection required

Storage

All data stored locally with proper encryption

The file scanning engine supports all Windows drive types (Fixed, Removable, Network, etc.) with multi-threaded scanning for improved performance. BAR includes validation of .bar files using signature verification and version compatibility checks.

Use Cases

Sensitive Document Sharing

Securely share confidential documents with colleagues or clients with automatic deletion after viewing

Temporary Credential Storage

Store passwords, API keys, or access tokens that self-destruct after use

Legal and Compliance

Meet data retention policies by ensuring files are automatically deleted after required periods

Personal Privacy

Protect sensitive personal information with files that can't be recovered after deletion

Secure Communication

Exchange sensitive information with built-in destruction capabilities

Corporate Environments

Protect intellectual property and trade secrets with controlled access

Healthcare Settings

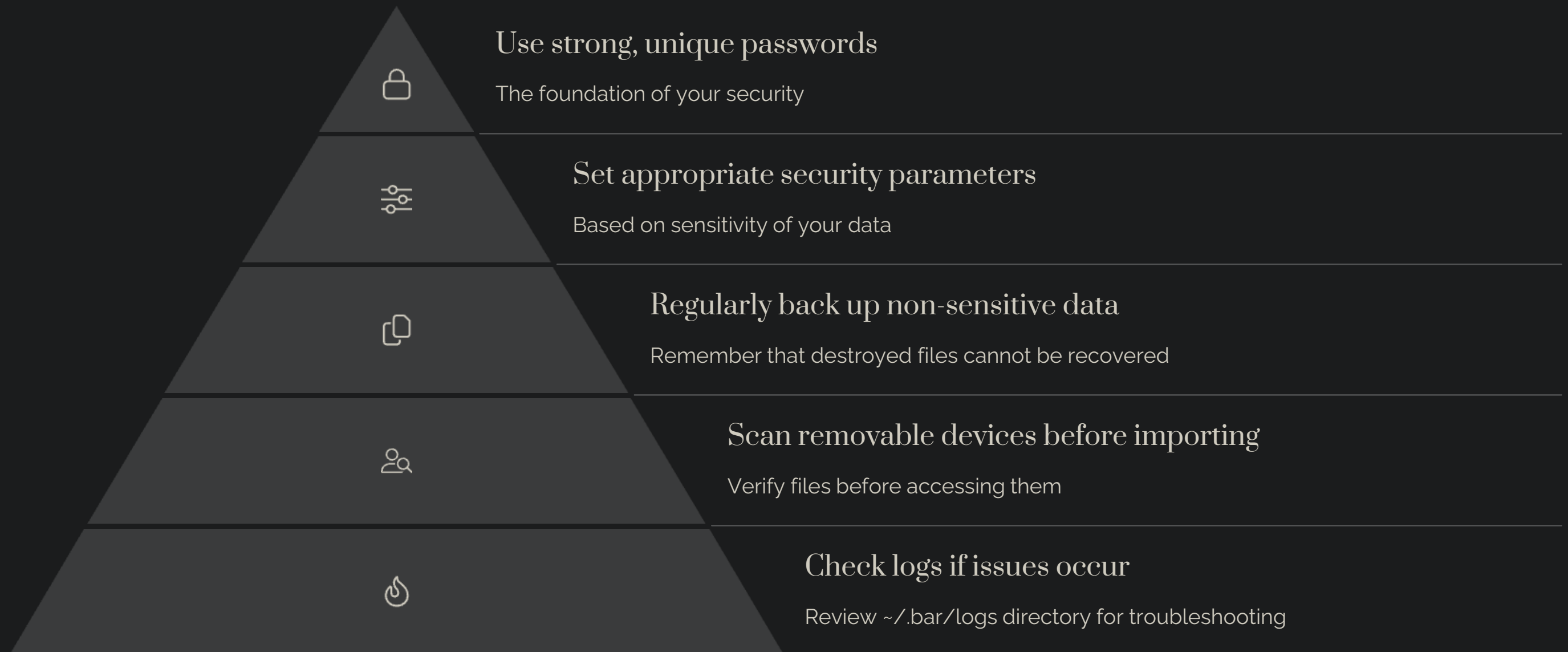Share patient information securely with automatic expiration

Financial Services

Protect financial documents and statements with time-limited access

Best Practices

Use strong, unique passwords

The foundation of your security

Set appropriate security parameters

Based on sensitivity of your data

Regularly back up non-sensitive data

Remember that destroyed files cannot be recovered

Scan removable devices before importing

Verify files before accessing them

Check logs if issues occur

Review ~/.bar/logs directory for troubleshooting

If you forget your password, your data cannot be recovered. This is a fundamental security feature of BAR that ensures your information remains protected.