Introduction

The **Cloud Donny Launcher** has been designed and developed by **Rhys Saddul** to accelerate the delivery of cloud projects by standardising deployment activities. The application is built on **Eduthing's standard solution designs**, ensuring that each environment is configured correctly, consistently, and in alignment with established best practices.

In addition to supporting day-to-day configuration, the launcher also provides tools to assist with **migration projects** and **new tenant setups**, helping to reduce complexity and improve accuracy during these critical phases.

The launcher provides a centralised interface with six core sections:

- **Azure**
- **Entra**
- **Intune**
- **OneDrive**
- **O365**
- **SharePoint**

Each section contains tools and scripts specific to its service area, enabling project teams to streamline configuration tasks and maintain adherence to design standards.

A **Help menu** is also available in the top left corner of the application. This provides access to:

- **Templates** required by the scripts contained within the application.
- The latest version of this **User Guide** for reference and operational support.

The following guide outlines each section of the application in detail and provides instructions for its use.

Key Benefits

Using the **Cloud Donny Launcher** provides the following advantages:

- **Speed** – Rapid deployment of cloud solutions with minimal manual intervention.
- **Consistency** – All environments are set up according to Eduthing's approved solution designs.
- **Compliance** – Adherence to standard configurations reduces the risk of misconfiguration.
- **Migration Readiness** – Built-in support for tenant setup and migration projects streamlines transitions.
- **Centralised Access** – A single interface for all major service areas, reducing the need to run individual scripts manually.

Context

Azure

Azure Scripts

Script Overview – Connected Cache Setup

⚠️ **Important:** This script must be run **on the server where the Connected Cache (sub-Linux system) will be installed**.

When you run this script, it:

1. Prompts you to enter the **school or trust code**, **Connected Cache VM name**, and **admin email address**.
2. Checks for required features such as **Hyper-V**, **WSL**, and **Azure CLI**, installing them if needed.
3. Verifies there is enough **free disk space** for Connected Cache.
4. Creates an **Azure resource group** if one doesn't already exist.
5. Creates the **Connected Cache resource** if it doesn't already exist.
6. Creates a **cache node** for the site if it doesn't already exist.
7. Displays the **Cache Node ID** and **DHCP options** needed to complete configuration.

Script Overview – Azure Storage Environment Setup

When you run this script, it:

1. Prompts you to enter the **school or trust code**.
2. Creates a **resource group** if one doesn't already exist.
3. Creates a **storage account** if one doesn't already exist, using standard settings.
4. Enables **7-day soft delete** for blobs and containers to protect against accidental deletion.
5. Creates a **blob container** if one doesn't already exist.
6. Sets up a **stored access policy** to allow secure access to the container.

Entra

Script Overview – Autopilot App Registration

When you run this script, it:

1. Registers a new **Azure app** named *AutopilotEnrollment*.
2. Creates a **service principal** for the app.
3. Assigns the required **Microsoft Graph API permissions**.
4. Disables the default *user_impersonation* permission scope.
5. Creates a **client secret** valid for two years.
6. Displays the **App ID**, **Client Secret**, and **Tenant ID** in a popup so they can be copied and saved securely.
7. Opens the **Azure Portal** and prompts you to **grant admin consent** for the app.

Script Overview – Exam Users Query Tool App Registration

When you run this script, it:

1. Launches a simple **GUI** for starting the registration process.
2. Registers a new **Azure app** called *Exam Users Query*.
3. Creates a **service principal** for the app.
4. Assigns the required **Microsoft Graph API permissions** (*GroupMember.Read.All* and *User.Read.All*).
5. Creates a **client secret** valid for two years.
6. Opens the **admin consent page** in Microsoft Edge for approval.
7. Displays the **App ID**, **Client Secret**, and **Tenant ID**, with an option to **copy them to the clipboard**.

This app is used to **query Entra ID (Azure AD) groups** so that **exam-related policies**, such as *disabling spell check* or *restricting internet access*, can be applied to specific user groups during exam sessions.

Script Overview – PaperCut Entra ID App Registration

When you run this script, it:

1. Launches a simple **GUI** to start the app registration process.
2. Registers a new **Azure app** named *PaperCut Entra ID Sync*.
3. Creates a **service principal** for the app.
4. Assigns the required **Microsoft Graph API permissions** (*User.Read*, *User.Read.All*, *Group.Read.All*, *GroupMember.Read.All*).

5. Sets up the **redirect URI** needed for PaperCut to authenticate.
6. Creates a **client secret** valid for two years.
7. Opens the **admin consent page** in Microsoft Edge to approve the app permissions.
8. Displays the **App ID**, **Client Secret**, and **Tenant ID**, with an option to **copy them to the clipboard**.

This app is used to allow **PaperCut** to synchronise with **Microsoft Entra ID (Azure AD)**, enabling user and group information to be imported automatically for print management and access control.

Intune

Script Overview – Android Mobile Groups Setup

When you run this script, it:

1. Prompts you to enter the **school code**, and optionally a **trust code**, then asks whether this is a **School** or **Trust** setup.
2. Creates all required **static and dynamic security groups** in **Entra ID (Azure AD)** for Android mobile device management.
3. Uses consistent **naming conventions** for both school and trust environments (e.g. *[EDU v2 Security] – SchoolCode All Corporate Android Mobile Devices*).
4. Configures **dynamic membership rules** based on device enrolment profiles.
5. Applies **group nesting**, linking school-level groups under trust-level groups where applicable.
6. Provides guidance on adding the **Intune Autopilot service principal** as an **owner** to key enrolment groups.

This script automates the full setup of **Android enrolment and management groups** in Entra ID, ensuring staff and student devices are properly organised for Intune policy assignment across schools and trusts.

Script Overview – Android Tablet Groups Setup

⚠️ **Important:** Microsoft Graph cannot automatically assign **service principal ownership** to groups. After running this script, you must **manually add the Intune Service Principal** as an **owner** to the groups listed in the output window.
The correct service principal should be either **“Intune Autopilot ConfidentialClient”** or **“Intune Provisioning Client”** — both are valid as long as the **AppID** is **f1346770-5b25-470b-88bd-d5744ab7952c**.

When you run this script, it:

1. Prompts you to enter the **school code**, and optionally a **trust code**, then asks whether this is a **School** or **Trust** setup.
2. Creates all required **static and dynamic security groups** in **Entra ID (Azure AD)** for Android tablet device management.

3. Uses consistent **naming conventions** for both school and trust environments (e.g. *[EDU v2 Security] – SchoolCode All Corporate Android Tablet Devices*).
4. Configures **dynamic membership rules** based on device enrolment profiles.
5. Applies **group nesting**, linking school-level groups under trust-level groups where applicable.
6. Lists the groups in the **output window** where the **Intune Service Principal** must be manually added as an **owner**.

This script automates the creation and structure of **Android tablet enrolment and management groups** in Entra ID, ensuring staff and student devices are correctly organised for Intune policy deployment across schools and trusts.

Script Overview – Exam Groups Setup

⚠️ **Important:** The groups created by this script — for example,
TrustCode Exam Disable Spellcheck → SchoolCode Exam Disable Spellcheck — are the ones that should be used in **Intune policies** to disable spell check during exams.
If you are setting up for a **trust**, use the **TrustCode** versions of the groups in your Intune configuration scripts; otherwise, use the **SchoolCode** groups.
This script works in conjunction with the **Exam Users Query Tool (Entra App)**, which queries these Entra groups to enforce exam-specific device restrictions.

When you run this script, it:

1. Prompts you to enter the **school code**, and optionally a **trust code**, then asks whether this is a **School** or **Trust** setup.
2. Creates the required **exam security groups** in **Entra ID (Azure AD)** for managing student exam conditions such as enabling or disabling spellcheck, restricting internet access, and grouping all exam students.
3. Uses clear, standardised **naming conventions** (e.g. *SchoolCode Exam Enable Spellcheck*, *TrustCode Exam All Students*).
4. For **trust-level setups**, automatically **nests school-level groups** under the corresponding trust-level groups (e.g. *TrustCode Exam Disable Spellcheck → SchoolCode Exam Disable Spellcheck*).
5. Displays the group creation and nesting results in the console for easy tracking.

This script automates the setup of **exam-related Entra ID security groups**, ensuring consistent and scalable management of **exam-specific Intune policies** such as spellcheck control and internet lockdowns across schools and trusts.

Script Overview – iPad Groups Setup

When you run this script, it:

1. Prompts you to enter the **school code**, and optionally a **trust code**, then asks whether this is a **School** or **Trust** setup.

2. Creates all required **static and dynamic security groups** in **Entra ID (Azure AD)** for iPad device management.
3. Uses consistent **naming conventions** for both school and trust environments (e.g. *[EDU v2 Security] – SchoolCode All Corporate iPad Devices*).
4. Configures **dynamic membership rules** based on device enrolment profiles (e.g. Staff Shared, Student Personal).
5. Applies **group nesting**, linking school-level groups under trust-level groups where applicable.
6. Lists the groups in the **output window** where the **Intune Service Principal** must be manually added as an **owner**.

This script automates the creation and structure of **iPad enrolment and management groups** in Entra ID, ensuring staff and student devices are correctly organised for Intune policy deployment across schools and trusts.

Script Overview – Mac Groups Setup

When you run this script, it:

1. Prompts you to enter the **school code**, and optionally a **trust code**, then asks whether this is a **School** or **Trust** setup.
2. Creates all necessary **static and dynamic Entra ID (Azure AD) groups** used for managing **Mac device enrolments**.
3. Uses consistent naming conventions (e.g. *[EDU v2 Security] – SchoolCode All Corporate Mac Devices*) for easier identification and standardisation across schools and trusts.
4. Creates **dynamic membership groups** based on the assigned **enrolment profiles** (e.g. Staff Shared, Student Personal).
5. Applies **nested group relationships**, automatically linking school-level groups under their trust-level equivalents where applicable.
6. Displays the resulting group structure and nesting relationships directly in the PowerShell output window.

This script automates the configuration of **Mac device management groups** within Entra ID, ensuring that **Intune policies** such as enrolment and configuration profiles are properly organised and scoped to the correct trust or school.

Script Overview – Windows Devices Groups Setup

⚠️ **Important:**

If you are using **AutoPilot v2**, you **must** ensure that the **Intune Autopilot ConfidentialClient** or **Intune Provisioning Client** (AppID above) is **added as an owner** to the relevant groups before proceeding.

Microsoft Graph cannot automatically assign **service principal ownership** to groups.
After running this script, you must **manually add the "Intune Autopilot ConfidentialClient" or "Intune**

Provisioning Client" service principal as an **owner** to the groups listed in the output.
Both are valid as long as the **AppID** is **f1346770-5b25-470b-88bd-d5744ab7952c**.

If you are using **AutoPilot v2**, you **must** ensure that the **Intune Autopilot ConfidentialClient** or **Intune Provisioning Client** (AppID above) is **added as an owner** to the relevant groups before proceeding.

When you run this script, it:

1. Prompts for the **school code** and optionally a **trust code**, then determines whether this is a **School** or **Trust** setup.
2. Creates all required **static and dynamic security groups** in **Entra ID (Azure AD)** for managing **Windows device configurations**.
3. Uses consistent **naming conventions** for easy identification (e.g. *[EDU v2 Security] – SchoolCode All Teacher Laptops*).
4. Builds **dynamic groups** based on device naming conventions (e.g. SchoolCode-TL- for Teacher Laptops).
5. Automatically applies **group nesting**, linking school-level groups under trust-level structures where applicable.
6. Includes **global software and assigned device groups** for managing **Intune app, configuration, and policy deployments**.

This script automates the creation and organisation of **Windows device management groups** within Entra ID, ensuring a standardised and scalable foundation for **Intune policy management** across schools and trusts.

O365

Script Overview – Export Aliases

This script reads a list of **user principal names (UPNs)** from an input CSV file and retrieves mailbox details from Exchange Online.

It collects and formats:

- **Primary email aliases** (SMTP: – uppercase prefix)
- **Secondary email aliases** (smtp: – lowercase prefix)

The output is exported to a CSV file containing each user's:

- Display Name
- User Principal Name
- Capital SMTP Aliases
- Lowercase SMTP Aliases

This script is useful for **auditing and verifying email alias configurations** across user mailboxes before migrations or configuration changes.