newline LAUNCH CONTROL

Launch Control technical whitepaper

Thursday, September 23, 2021

Introduction

About the software developer

DisplayNote is a technology company established in 2012, whose mission is to simplify how people present, share and collaborate on content.

DisplayNote products are primarily found in education and business spaces such as meeting rooms, classrooms, huddle spaces, training rooms, etc.

With offices in the UK and in Spain, the company's multi-cultural workforce understands the importance of communication without barriers.

DisplayNote prides itself as a technology company that delivers high-performing software with unrivalled user experience.

For more information about DisplayNote, visit displaynote.com.

About the product

Launch Control is a software solution designed to simplify interactions Windows systems found within shared spaces.

The primary software component is available on Windows 10 for use on devices within shared spaces, while its client software counterpart is available on iOS and Android OS for use on smartphones.

The software's core functionality includes:

- MS Teams and Zoom integration for joining calls with one touch. *
- Intuitive user interface as an overlay to the Windows desktop.
- Kiosk mode to prevent access to the Windows system.
- Exchange and Google integration for room calendar configurations.
- Microsoft 365 authentication for access to personal content.

*Full list of supported video conferencing tools also includes GoToMeeting, Google Meet, Lifesize, Bluejeans, Webex, Skype for Business

Requirements

Launch Control Windows

- Minimum resolution: 1080p
- Operating system: Windows 10 version 1809 (10.0; Build 17763). Launch Control will not install on any earlier versions of Windows.
- Network configuration: Ethernet or Wi-Fi (allow TCP connection to remote port 5671)
    - Just default web ports (80 and 443) and TCP connections to remote port 5671.
    - The full table of required protocols, domains and ports are below:

Protocol	URL	Port
amqp/ssl	heavy-duck.rmq.cloudamqp.com	5671
https	displaynoteshadow.azurewebsites.net/api	443
https	launcher-service.azurewebsites.net/api	443
https	displaynote.count.ly	443
https	graph.microsoft.com	443
https	outlook.office365.com	443
https	googleapis.com	443

- Network conditions: This is not a high demanding application. As such, it can be operated on low-speed networks.

Supported resolution combinations

- (4K) 3840x2160 at 150%, 200% and 300% scaling
- (2K) 2560x1440 at 100%, 150% and 200% scaling
- (Full HD) 1920x1080 at 100% and 150% scaling

iOS app

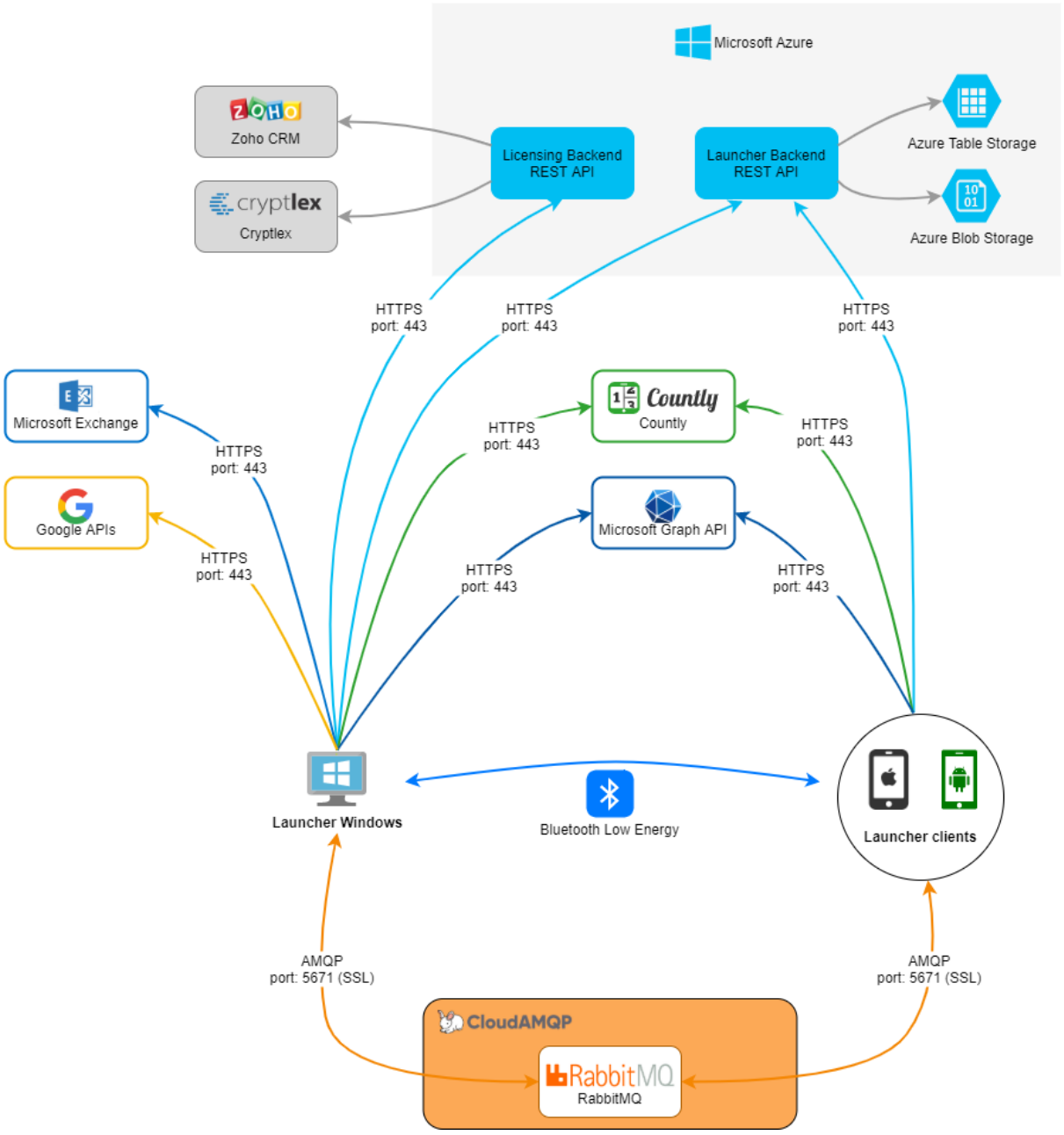
- iPhones with iOS 12.1
- Network configuration: Ethernet, 3G or Wi-Fi
- Network conditions: This is not a high demanding application. As such, it can be operated on low-speed networks.

Android app

- Android devices with Android 5.0
- Network configuration: Ethernet, 3G or Wi-Fi
- Network conditions: This is not a high demanding application. As such, it can be operated on low-speed networks.

Architecture

The following diagram illustrates the Launch Control architecture between Launch Control receivers, clients, the message broker and backends. Please see additional information in subsequent sections.



Please note that any reference to **Launcher** in the diagram above, refers to **Launch Control.**

Launch Control receiver

Launch Control receiver for Windows is the central piece of software for the Launch Control solution. It uses Bluetooth Low Energy (BLE) to announce its presence to Launcher client apps.

Once a Launcher client app (see information in next section) is detected, all communication between them is achieved over RabbitMQ hosted by CloudAMQP, secured by TLS.

Launcher client app

The Launcher client app (iOS/Android) is a mobile app that acts as user identifier and allows users to interact with Launch Control receivers. The iOS version of the mobile app uses Bluetooth Low Energy to discover nearby devices that are running the Launch Control application.

Once a Launch Control receiver is detected, all communication between them is done over RabbitMQ hosted by CloudAMQP, secured by TLS.

Backends at Microsoft Azure

Our backends are hosted on Microsoft Azure. Azure's data centres are geographically dispersed and comply to ISO/IEC 20000-1:2018, ISO 22301:2019, ISO/IEC 27001:2013, ISO/IEC 27017:2015, ISO/IEC 27018:2019, ISO/IEC 27701:2019, ISO 9001:2015, SOC 1, SOC 2 and SOC 3 and have a CSA STAR certification. The backend manages licensing and updates API REST, and it is only accessed by the Launch Control receiver application.

Encryption and certificates

Communication between our backends and the Launch Control receivers or clients, is encrypted and transmitted over HTTPS with 2048-bit asymmetric encryption and 256-bit symmetric encryption, using certificates from third party credited authorities.

Launch Control backend

Launch Control backend is accessed by Launch Control receivers and clients, it is used for verifying that devices discovered through BLE are Launch Control instances, and returns all information needed to establish a connection via RabbitMQ.

Licensing backend

Launch Control checks licenses through the licensing backend, which uses Zoho CRM and Cryptlex to manage user information and licenses.

Countly

Countly is a third-party service used for analytics purposes.

Microsoft Graph API

User calendars and OneDrive items are retrieved using the Microsoft Graph REST API.

Microsoft Exchange

Microsoft Exchange is a web service used to retrieve the Microsoft accounts room calendars.

Google APIs

Google APIs is a web service used to retrieve the Google accounts room calendars.

Launching calls

Displaynote Launch Control receiver can start any meeting/call that has been set up through Microsoft Exchange, as each calendar event contains a public link to the event that Launch Control uses to start the call.

Microsoft Teams

If MS Teams is not installed on the Windows device, the Launch Control receiver will open the meeting link via web browser. Otherwise, the Launch Control receiver will call MS Teams process passing the meeting link a parameter. Launch Control receiver does not sign in or out of MS Teams accounts. If MS Teams is previously signed in with an account, user will join the meeting/call under that account.

Skype for Business (deprecated)

If Skype for Business (SfB) is not installed on the Windows device, the Launch Control receiver will open the meeting link via web browser. Otherwise, the Launch Control receiver will interact with Skype for Business using its SDK.

The SfB SDK needs the username and password to sign into Skype for Business. This information is requested from the user via the Launch Control client app and it is sent to the Launch Control receiver securely through RabbitMQ using TLS. The password is never stored and it is completely forgotten once users disconnect or the application is closed.

Skype for Business support will be removed soon.

Zoom

If a Zoom link is detected and the native Zoom app is detected on the system, the meeting will be launched on the native app. Otherwise, the meeting link will be launched via web browser.

GoToMeeting

If a GoToMeeting link is detected and the native GoToMeeting app is detected on the system, the meeting will be launched on the native app. Otherwise, the meeting link will be launched via web browser.

Webex

If a Webex link is detected and the native Webex app is detected on the system, the native app will be launched but the meeting won't be launched automatically. Otherwise, the meeting link will be launched via web browser.

Other solutions

Launch Control also detects meeting links for Zoom, GoToMeeting, Bluejeans, Webex, Google Meet and Lifesize. Once a meeting link is detected for any of these systems, Launch Control can start the call opening the link in the system default browser, the browser will delegate to native apps when possible.

Data caching and clearing cache

Cleanup triggers

Launch Control clears user data by session. A session in Launch Control is finished when any of these conditions are met:

- Launch Control application is closed, or the system is shutdown.
- User manually presses ‘Cleanup’ button from the home screen.
- A user is connected/disconnected through Launcher mobile apps.
- A user manually signs in/out on Launch Control using a Microsoft account.
- An automatic Cleanup been set (on meeting end; after a period of inactivity; at a scheduled time).

When a Cleanup operation is run, the data is cleared based on one of two scenarios: Win32 Apps and UWP Apps (Windows store apps). See information below.

Win32 apps

Launch Control executes regular Windows apps under a private and temporary Windows user. This means that anytime a Cleanup trigger (listed above) occurs, all data for these applications and processes are removed from the system. We can say that regular Windows applications run in a private session.

There is an exception for applications installed under user folder (C:\Users\CurrentUser). Applications installed under the current user cannot be executed by other Windows users, and therefore those applications are not able to run in private session. These applications will still be launched but they will not be subject to any effects of a Cleanup operation. While they are connected to the Launch Control receiver, Launcher mobile app users will see a warning message when they launch an application installed under the current user.

UWP apps (Windows store apps)

This kind of applications cannot be run under a temporary user, but they have a structured way to store information. Anytime one of the Cleanup triggers (listed above) occurs, the following steps are executed:

1. Content of following app folders is deleted: Settings, AppData, LocalState, LocalCache, AC.
2. Windows credentials are deleted.

Device discovery

Launcher smartphone application includes of a device discovery mechanism to establish a connection with the Launch Control Windows software application. This proximity detection function is achieved using Bluetooth Low Energy (BLE)

Launch Control receiver

Launch Control receiver publishes BLE advertisement packets with a unique Launch Control identifier, which is refreshed from our backend in each execution of the Launch Control receiver. Note that Bluetooth must be enabled on the Windows device for the BLE function of Launch Control receiver to operate.

Launcher mobile app

The Launcher mobile app (iOS) listens for BLE advertisement packets, and when a Launch Control packet is found the app retrieves all information associated with that Launch Control identifier form the backend. If a valid Launch Control packet is found, it will appear as a nearby device on the Launcher mobile app.

Proximity limit

If the Launcher mobile app is beyond the proximity limit of a Launch Control receiver, it will not receive packets and therefore it will be able to discover that Launch Control receiver.

Additionally, the Launcher mobile app will automatically disconnect from a Launch Control receiver upon exceeding the proximity limit.

Location services

For the Launcher mobile app (iOS), users need to allow requests in order to discover nearby Launch Control receivers.

Location services option – Always allow

The recommended option is ‘Always allow’, which will discover devices when app is not running, or when the iOS device is locked. When ‘Always allow’ is enabled, a notification will appear for each device discovered.

Location services option – Only while using the app

The option to allow locations services ‘Only while using the app’ will only discover devices while the app is running in the foreground. Sending the app to the background will be managed as a device that is losing its connection.

Integrations and authentication

Microsoft Exchange integration

Launch Control receiver can be configured to connect with an MS Exchange account to retrieve calendar events, including room accounts if they are properly set up.

MS Exchange integration has been achieved using MS Exchange Web Services API (see Exchange Online and Exchange development). We support two kinds of authentication:

1. Basic authentication: MS Exchange passwords are encrypted and stored in Launch Control receivers' application data.
2. Modern authentication (OAuth): Authentication is done using the Microsoft identity platform MS Identity Client documentation. No passwords are stored by Launch Control.

Authentication with Microsoft 365

Users can sign in within their Microsoft 365 account to get Outlook calendar and OneDrive information.

Authentication is done using the Microsoft identity platform to retrieve a valid token that is used to access Microsoft Graph API.

As authentication is done in Launcher client apps, the Microsoft token is sent to Launch Control receiver securely through RabbitMQ using TLS. Authentication tokens are never stored but are placed in the application's temporary memory space, which never persists to the file system memory. This token is completely forgotten once the user disconnects, or the application is closed.

Graph API Permission scopes

Launch Control requires the following set of permissions to be granted by the user or administrator to access Outlook calendar and OneDrive information. To find out more, go to Microsoft Graph permissions reference.

- **User.Read**: Allows users to sign into the app and allows the app to read the profile of signed in users. It also allows the app to read basic company information of signed in users.
- **User.ReadBasic.All**: Allows the app to read a basic set of profile properties of other users in your organisation on behalf of the signed in user. This includes display name, first and last name, email address, open extensions, and photo. Also allows the app to read the full profile of the signed-in user.
- **Calendars.Read**: Allows the app to read events in user calendars.
- **Files.Read:** Allows the app to read the signed-in user's files.

Google Calendar integration

Launch Control receiver can be configured to connect with a Google account to retrieve its calendar events. Google calendar integration has been achieved using Google APIs [Google APIs calendar documentation](#).

Kiosk mode

Launch Control receiver includes an optional function called ‘Kiosk mode’, which secures the Windows system. When Launch Control’s dedicated kiosk mode is enabled, the following behaviour occurs:

- Windows desktop is not accessible
- Windows task bar is hidden
- Launch Control receiver automatically runs when the system boots up.
- Launch Control runs in full screen.
- Users cannot switch between application windows using keyboard or gesture commands.
- Users cannot execute any Ctrl or Alt key shortcut
- The following Windows features are disabled: cmd, running batch files, settings, control panel, task manager, registry, auto-open explorer when external drive is inserted.

Information collected

Depending on your use of our Services, we may collect some or all of the following personal data such as; Name, Date of birth, Company name, Job title, contact information such as email addresses and telephone numbers, demographic information such as post code, preferences, and interests, IP address, operating system.

Please review our privacy policy for more information regarding capturing of personal information.

If you have any issues or further questions, contact support@displaynote.com