BEHAVIOURAL BIOMETRICS

Why device isn't enough

A device ID tells you the device. Behavioural biometrics tells you what is happening in the session.

A device ID tells you which device is connecting. It does not tell you who is in control of it, or whether the session is driven by a person, a bot, or a remote attacker. Fraud farms run real phones that every device-intelligence vendor will correctly read as separate devices. A scam victim acts on their own trusted device, so they pass MFA and device checks. Behavioural biometrics closes this gap. It reads how the device is held, touched, typed on and moved through the journey, and combines those signals to reveal risk that device recognition on its own will miss.

HOW IT'S CAPTURED

Darwinium's JavaScript tag on web and native mobile SDK on Android and iOS pull behavioural biometrics from the device's sensors and the user's inputs. Signals that are temporal are polled **roughly every 200 ms** during profiling and converted into statistics such as **avg, min, max, median and std_dev**, available on the event and in rules.

COVERAGE Behavioural biometrics coverage

Mouse WEB	**Strokes** Inflection, distance	**Timing** Speed, intervals, duration	**Clicks** Left, right, middle	**Navigation** Scrolls, off-page	
Touch MOBILE	**Swipes** Inflection, distance	**Timing** Speed, intervals, duration	**Tap features** Size, pressure	**Tap location** Place on screen	
Keyboard	**Navigation** Tabs, arrows, enter	**Typing cadence** Hesitancy, flight, dwell	**Corrections** Backspace, deletes	**Auto** Autofill, autocorrect	**Shortcuts** Copy, cut, paste
Sensor	**Acceleration** Magnitude, direction	**Orientation** Tilt, rotation	**Magnetic field** Magnitude, direction	**Luminance** Intensity, change	**Battery charge**

Mouse applies to web. On the mobile SDK, touch is its equivalent. Each cell is a family of signals Darwinium derives.

WORKED EXAMPLE A remote-access scam, seen through behaviour

Four signals on one high-value payment

ANONYMISED MALWARE CASE

The device looked trusted and passed every device check. The session did not. On the victim's own device, four signals fired together on one high-value payment:

- **Synthetic touch** — `touch_source = TOUCH_UNKNOWN`
- **Side-loaded package** — `HAS_SIDE_LOADED_PACKAGES`
- **Accessibility enabled** — `accessibility_enabled = true`
- **Live call audio** — `audio_device.is_playing`

Verdict. The sensors agreed: the phone lay flat and still throughout. Because the device ID was the victim's own, the payment passed MFA, so device-only checks see nothing wrong.



FIG 1 The same signals in the event table: is_playing, accessibility, side-loaded packages and touch_source set to TOUCH_UNKNOWN on the flagged payments.

PERMISSIONS Behavioural biometrics read without extra permissions

No special permission needed

These signals are available by default, because apps already receive keyboard input and orientation changes. **Darwinium never needs to raise its own permission popup**; it inherits what the app already has.

`touch and swipe` `sensors` `keyboard` `timings`

BEHAVIOURAL BIOMETRICS

The four genres of behavioural signal

Sensor, touch, timing and keyboard. What each one captures, and the risks it catches.

01 GENRE Sensor biometrics

Ambient sensors in the phone: movement, orientation, magnetic field, light, proximity and heading. Gravity across X, Y and Z reveals orientation, and the sign of the vector separates face-up from face-down. Android returns m/s² (9.81 = 1 g); iOS returns normalised G.

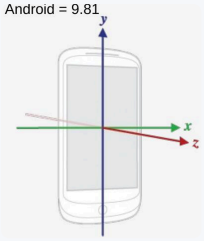

FIG 2 Gravity along X, Y, Z gives orientation.

ATTRIBUTES

GRAVITY_X/Y/Z · ACCELEROMETER · LINEAR_ACCEL · GYROSCOPE · MAGNETIC_FIELD · LIGHT · PROXIMITY · HEADING

EXAMPLE RULES AND THE RISK THEY CATCH

DEVICE_FLAT · DEVICE_UPRIGHT · NO_MOVEMENT

Catches device farms and emulators, which sit at fixed orientations with no real movement.

02 GENRE Touch and swipe biometrics

Taps and swipes on the screen. Taps carry area and pressure; swipes carry a full geometry of path length, straight-line distance, curvature and timing. The touch source shows whether input came from a finger or from software.

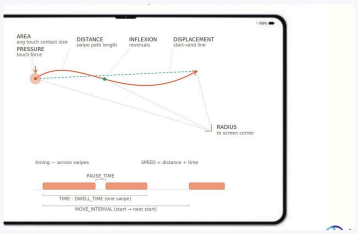

FIG 3 Distance, displacement, inflexion, radius and area of one swipe.

ATTRIBUTES

DISTANCE · DISPLACEMENT · INFLEXION · RADIUS · SPEED · AREA · PRESSURE · DWELL_TIME · touch_source

EXAMPLE RULES AND THE RISK THEY CATCH

STRAIGHT_SWIPE · TOUCH_SOURCE_UNKNOWN

Catches automation and remote-access malware, where swipes turn linear and inputs are not from a finger.

03 GENRE Timing and duration

The whole session under one journey, with a start timestamp and the time since each previous step. Fast completion points to automation or abuse; slow or hesitant behaviour can point to coercion.

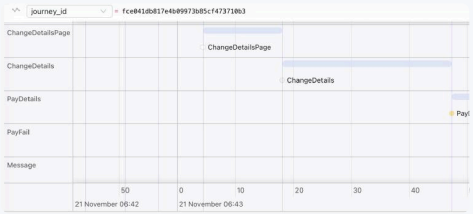

FIG 4 One session on a timeline, step by step.

ATTRIBUTES

biometric_activity_delay · biometric_activity_duration · biometric_max_hesitation · first_key_down · first_swipe_start

EXAMPLE RULES AND THE RISK THEY CATCH

FAST_DURATION · HIGH_HESITATION

Flags bot speed in registration or payment, and hesitation that can indicate a scam.

04 GENRE Keyboard biometrics

Virtual-keyboard interaction captured per field, with the field context (LOGIN, FIRST_NAME and so on) set in the app at profiling start. Behaviour is context dependent, so a paste is far likelier in an ID field than in a first name. Field contents are never collected.

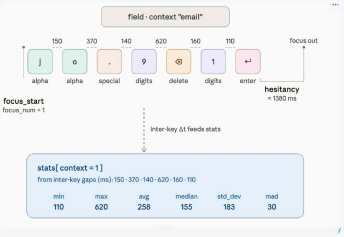

FIG 5 Per-field capture of key counts, hesitancy and timing.

ATTRIBUTES

alpha · digits · special · enter · delete · focus_num · focus_start · hesitancy

EXAMPLE RULES AND THE RISK THEY CATCH

PASTE_DETECTED · RANDOM_FOCUS_ORDER

Flags synthetic input and agentic AI, which paste sensitive fields and fill them out of order.