별첨. 월간 로그 점검 보고서

(로그 보관 및 관리 정책 v1.3 제 4 장에 따른 표준 양식)

점검일	2026 년 8 월 1 일	점검자	이효영	검토자 (보안책임자)	[illegible]
종합판정	정상 ☑ 경고 □ 이상 □	수시점검 사유	-	소급점검 여부	Y □ / N ☑

■ 작성 방법

각 점검 항목의 쿼리는 당월 건수·전월 대비 증감·누락일수·판정까지 한 번에 계산하여 결과가 표 형태로 나온다. ① 쿼리 첫 줄의 SET @m 값만 점검 대상 월의 1 일로 수정하고 ② 전체를 실행한 뒤 ③ 결과 그리드 화면을 그대로 캡처하여 [캡처 붙여넣기] 칸에 첨부한다. ④ 쿼리가 계산한 판정을 확인하여 체크하고, 경고·이상 시 ④후속조치 기록에 내용을 남긴다.

※ 결과 캡처에는 실행한 쿼리와 결과 그리드가 함께 보이면 더 좋다(DBeaver·HeidiSQL 등). 컬럼·상태값·임계치는 운영 스키마 기준으로 필요 시 조정하고, 조정 시 사유를 남긴다.

※ 본 보고서의 캡처는 점검 대상 월 기준이다(가이드라인 4.3 로그 점검 대응). 이와 별개로 인증 심사 증빙용「모든 로그 저장 기록(로그별 저장 파일 전체 목록 캡처)」은 보관 중인 전체 기간 기준으로 별도 준비한다(가이드라인 4.1·4.2 로그 보관 대응).

① 이용자 로그 점검

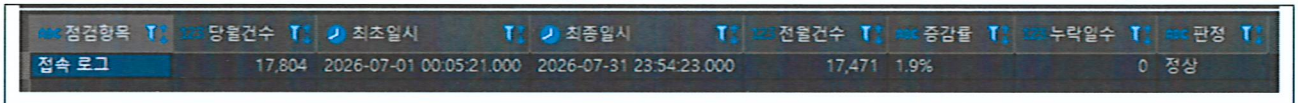
1. 접속 로그

점검대상	MJ_LOGIN_LOG, HACK_IP	저장위치	DB	보관기간	1 년 이상

▸ **점검 SQL (실행 후 결과 그리드를 그대로 캡처)**

```
SET @m := '2026-08-01';  -- ★ 점검 대상 월의 1일 (이 줄만 수정)
SELECT '접속 로그' AS `점검항목`,
       t.cur AS `당월건수`, t.first_dt AS `최초일시`, t.last_dt AS `최종일시`,
       t.prev AS `전월건수`,
       CONCAT(IFNULL(ROUND((t.cur - t.prev) / NULLIF(t.prev, 0) * 100, 1), '-'), '%') AS `증감률`,
       DAY(LAST_DAY(@m)) - t.days AS `누락일수`,
       CASE WHEN t.cur = 0                          THEN '이상(미생성)'
            WHEN DAY(LAST_DAY(@m)) > t.days         THEN '경고(누락의심)'
            WHEN t.prev > 0 AND t.cur > t.prev*2 THEN '경고(급증)'
            ELSE '정상' END AS `판정`
FROM (SELECT SUM(LOGIN_DT >= @m)                          AS cur,
             SUM(LOGIN_DT <  @m)                          AS prev,
             MIN(CASE WHEN LOGIN_DT >= @m THEN LOGIN_DT END) AS first_dt,
             MAX(CASE WHEN LOGIN_DT >= @m THEN LOGIN_DT END) AS last_dt,
             COUNT(DISTINCT CASE WHEN LOGIN_DT >= @m THEN DATE(LOGIN_DT) END) AS days
       FROM MJ_LOGIN_LOG
      WHERE LOGIN_DT >= DATE_SUB(@m, INTERVAL 1 MONTH)
        AND LOGIN_DT <  DATE_ADD(@m, INTERVAL 1 MONTH)) t;
```

▸ **결과 캡처**



점검항목	당월건수	최초일시	최종일시	전월건수	증감률	누락일수	판정
접속 로그	17,804	2026-07-01 00:05:21.000	2026-07-31 23:54:23.000	17,471	1.9%	0	정상

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

2. 인증 로그 (회원가입·본인확인)

점검대상	MJ_CERT_LOG, LETTNGNRLMBER_CERT_LOGIN_LOG_IP	저장위치	DB	보관기간	1 년 이상

▸ **점검 SQL (실행 후 결과 그리드를 그대로 캡처)**

```
SET @m := '2026-08-01';  -- ★ 점검 대상 월의 1일 (이 줄만 수정)
SELECT '인증 로그' AS `점검항목`,
       t.cur AS `당월건수`, t.ok AS `성공`, t.ng AS `실패`, t.prev AS `전월건수`,
       CONCAT(IFNULL(ROUND((t.cur - t.prev) / NULLIF(t.prev, 0) * 100, 1), '-'), '%') AS `증감률`,
       CASE WHEN t.cur = 0                          THEN '확인필요(0건)'
            WHEN t.prev > 0 AND t.cur > t.prev*2 THEN '경고(급증)'
            ELSE '정상' END AS `판정`
FROM (SELECT SUM(REGDATE >= @m) AS cur,
             SUM(REGDATE <  @m) AS prev,
             SUM(REGDATE >= @m AND CERT_RESULT =  'Y') AS ok,
             SUM(REGDATE >= @m AND CERT_RESULT <> 'Y') AS ng
```

```
       FROM MJ_CERT_LOG
      WHERE REGDATE >= DATE_SUB(@m, INTERVAL 1 MONTH)
        AND REGDATE <  DATE_ADD(@m, INTERVAL 1 MONTH)) t;
-- ※ CERT_RESULT 성공값('Y')은 운영 코드 기준으로 조정
```

▸ **결과 캡처**

점검항목	당월건수	성공	실패	전월건수	증감률	판정
인증 로그	2,352	781	7	2,975	-20.9%	정상

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

2-1. 추가인증 로그 (2 차 인증 · 신설 SECOND_AUTH_LOG)

점검대상	SECOND_AUTH_LOG (2 차 인증 로그 · 2026-07 신설)	**저장위치**	DB	**보관기간**	1 년 이상

▸ **점검 SQL (실행 후 결과 그리드를 그대로 캡처)**

```
SET @m := '2026-08-01';  -- ★ 점검 대상 월의 1 일 (이 줄만 수정)
SELECT '추가인증(2 차) 로그' AS `점검항목`,
       COUNT(*) AS `당월건수`,
       SUM(LOGIN_RESULT = 'SUCCESS') AS `성공`, SUM(LOGIN_RESULT = 'FAIL') AS `실패`,
       COUNT(DISTINCT TRY_LOGIN_ID) AS `시도계정수`,
       MIN(FRST_REGIST_PNTTM) AS `최초일시`, MAX(FRST_REGIST_PNTTM) AS `최종일시`,
       CASE WHEN COUNT(*) = 0 THEN '확인필요(0 건)' ELSE '정상' END AS `판정`
  FROM SECOND_AUTH_LOG
 WHERE FRST_REGIST_PNTTM >= @m
   AND FRST_REGIST_PNTTM < DATE_ADD(@m, INTERVAL 1 MONTH);
-- ※ 컬럼: LOGIN_RESULT(성공/실패)·AUTH_STEP(인증단계)·TRY_LOGIN_ID(시도계정)·REASON·ACCESS_IP·MSG_ID
등
-- ※ LOGIN_RESULT: SUCCESS/FAIL · AUTH_STEP: SECONDARY(2 차 인증)
-- ※ 적재 개시 월부터 점검 적용, 전월 대비 비교는 2 개월차부터 기재
```

▸ **결과 캡처**

점검항목	당월건수	성공	실패	시도계정수	최초일시	최종일시	판정
추가인증(2차) 로그	0	[NULL]	[NULL]	0	[NULL]	[NULL]	확인필요(0건)

※ 신규 생성 테이블이라 코드 운영 반영 후 데이터 저장됨

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

3. 문자 발송 이력

점검대상	MJ_MSG_GROUP_DATA, MJ_MSG_DATA	저장위치	DB	보관기간	1 년 이상

▸ **점검 SQL (실행 후 결과 그리드를 그대로 캡처)**

```
SET @m := '2026-08-01';  -- ★ 점검 대상 월의 1 일 (이 줄만 수정)
SELECT '문자 발송 이력' AS `점검항목`,
       m.cur AS `발송건수(당월)`, g.cur AS `그룹건수(당월)`, m.prev AS `발송건수(전월)`,
       CONCAT(IFNULL(ROUND((m.cur - m.prev) / NULLIF(m.prev, 0) * 100, 1), '-'), '%') AS `증감률`,
       DAY(LAST_DAY(@m)) - m.days AS `누락일수`,
       CASE WHEN m.cur = 0                          THEN '이상(미생성)'
            WHEN DAY(LAST_DAY(@m)) > m.days         THEN '경고(누락의심)'
            WHEN m.prev > 0 AND m.cur > m.prev*2 THEN '경고(급증)'
            ELSE '정상' END AS `판정`
FROM (SELECT SUM(REQ_DATE >= @m) AS cur,
             SUM(REQ_DATE <  @m) AS prev,
             COUNT(DISTINCT CASE WHEN REQ_DATE >= @m THEN DATE(REQ_DATE) END) AS days
        FROM MJ_MSG_DATA
       WHERE REQ_DATE >= DATE_SUB(@m, INTERVAL 1 MONTH)
         AND REQ_DATE <  DATE_ADD(@m, INTERVAL 1 MONTH)) m,
     (SELECT COUNT(*) AS cur
        FROM MJ_MSG_GROUP_DATA
       WHERE REQ_DATE >= @m AND REQ_DATE < DATE_ADD(@m, INTERVAL 1 MONTH)) g;
-- ※ 일자 컬럼은 운영 스키마 기준(REQ_DATE/SENT_DATE)으로 조정
```

▸ **결과 캡처**

점검항목	발송건수(당월)	그룹건수(당월)	발송건수(전월)	증감률	누락일수	판정
문자 발송 이력	1,811,189	43,302	1,828,446	-0.9%	0	정상

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

4. 발송 이력 (팩스·카카오)

점검대상	PGI_FAXTRAN, MJ_MSG_DATA(MSG_TYPE 8·9)	저장위치	DB	보관기간	1 년 이상

▸ **점검 SQL (실행 후 결과 그리드를 그대로 캡처)**

```
SET @m := '2026-08-01';  -- ★ 점검 대상 월의 1 일 (이 줄만 수정)
SELECT '팩스 발송' AS `점검항목`,
       SUM(SendDate >= @m) AS `당월건수`, NULL AS `당월건수 2`,
       SUM(SendDate <  @m) AS `전월건수`,
       CASE WHEN SUM(SendDate >= @m) = 0 THEN '확인필요(0 건)' ELSE '정상' END AS `판정`
  FROM PGI_FAXTRAN
 WHERE SendDate >= DATE_SUB(@m, INTERVAL 1 MONTH)
   AND SendDate <  DATE_ADD(@m, INTERVAL 1 MONTH)
UNION ALL
SELECT '카카오 발송(알림톡·친구톡)',
       SUM(REQ_DATE >= @m AND MSG_TYPE = 8),   -- 알림톡(당월)
       SUM(REQ_DATE >= @m AND MSG_TYPE = 9),   -- 친구톡(당월)
       SUM(REQ_DATE <  @m),                    -- 전월합계
       CASE WHEN SUM(REQ_DATE >= @m) = 0 THEN '확인필요(0 건)' ELSE '정상' END
  FROM MJ_MSG_DATA
 WHERE MSG_TYPE IN (8, 9)   -- 8: 카카오 알림톡, 9: 카카오 친구톡
```

```
   AND REQ_DATE >= DATE_SUB(@m, INTERVAL 1 MONTH)
   AND REQ_DATE <  DATE_ADD(@m, INTERVAL 1 MONTH);
-- ※ 카카오 발송은 별도 테이블이 아니라 MJ_MSG_DATA 에 MSG_TYPE(8·9)으로 구분 저장됨
-- ※ 팩스 미사용 시 팩스 행은 0 건 → '확인필요(0 건)'로 표시되며, 조치·비고에 '미사용' 기재
```

▸ **결과 캡처**

점검항목	당월건수	당월건수2	전월건수	판정
팩스 발송	40	[NULL]	44	정상
카카오 발송(알림톡·친구톡)	203,048	11	68,195	정상

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

5. 과금·결제 이력

점검대상	MJ_PG, MJ_CASH, MJ_POINT, MJ_TAX, KGM_PG	**저장위치**	DB	**보관기간**	5 년

▸ **점검 SQL (실행 후 결과 그리드를 그대로 캡처)**

```
SET @m := '2026-08-01';  -- ★ 점검 대상 월의 1 일 (이 줄만 수정)
SELECT '과금·결제' AS `점검항목`,
       t.cur AS `당월건수`, FORMAT(t.amt, 0) AS `당월금액`, t.prev AS `전월건수`,
       CONCAT(IFNULL(ROUND((t.cur - t.prev) / NULLIF(t.prev, 0) * 100, 1), '-'), '%') AS `증감률`,
       c.cur AS `캐시변동(당월)`,
       CASE WHEN t.cur = 0                        THEN '확인필요(0 건)'
            WHEN t.prev > 0 AND t.cur > t.prev*2 THEN '경고(급증)'
            ELSE '정상' END AS `판정`
FROM (SELECT SUM(REG_DATE >= @m) AS cur,
             SUM(REG_DATE <  @m) AS prev,
             SUM(CASE WHEN REG_DATE >= @m THEN AMT ELSE 0 END) AS amt
        FROM MJ_PG
       WHERE REG_DATE >= DATE_SUB(@m, INTERVAL 1 MONTH)
         AND REG_DATE <  DATE_ADD(@m, INTERVAL 1 MONTH)) t,
     (SELECT COUNT(*) AS cur
        FROM MJ_CASH
       WHERE FRST_REGIST_PNTTM >= @m
         AND FRST_REGIST_PNTTM < DATE_ADD(@m, INTERVAL 1 MONTH)) c;
-- ※ MJ_CASH 의 일자 컬럼은 FRST_REGIST_PNTTM (REG_DATE 아님)
```

▸ **결과 캡처**

	점검항목	당월건수	당월금액	전월건수	증감률	캐시변동(당월)	판정
1	과금·결제	1,013	72,722,721	1,093	-7.3%	135,413	정상

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

② 시스템 로그 점검

1. 관리자 접속·행위

점검대상	LETTNADMINLOG + lettnemplyrinfo (관리자 로그 화면과 동일 기준)	저장위치	DB	보관기간	1 년 이상

▸ **점검 SQL (실행 후 결과 그리드를 그대로 캡처)**

```
SET @m := '2026-08-01';  -- ★ 점검 대상 월의 1 일 (이 줄만 수정)
SELECT '관리자 접속·행위' AS `점검항목`,
       t.cur AS `당월건수`, t.admins AS `접속 관리자수`,
       t.first_dt AS `최초일시`, t.last_dt AS `최종일시`, t.prev AS `전월건수`,
       CONCAT(IFNULL(ROUND((t.cur - t.prev) / NULLIF(t.prev, 0) * 100, 1), '-'), '%') AS `증감률`,
       CASE WHEN t.cur = 0                        THEN '이상(미생성)'
            WHEN t.prev > 0 AND t.cur > t.prev*2 THEN '경고(급증)'
            ELSE '정상' END AS `판정`
FROM (SELECT SUM(CREAT_DT >= @m) AS cur,
             SUM(CREAT_DT <  @m) AS prev,
             COUNT(DISTINCT CASE WHEN CREAT_DT >= @m THEN CONECT_ID END) AS admins,
             MIN(CASE WHEN CREAT_DT >= @m THEN CREAT_DT END) AS first_dt,
             MAX(CASE WHEN CREAT_DT >= @m THEN CREAT_DT END) AS last_dt
        FROM LETTNADMINLOG
       WHERE CONECT_ID != ''
         AND CREAT_DT >= DATE_SUB(@m, INTERVAL 1 MONTH)
         AND CREAT_DT <  DATE_ADD(@m, INTERVAL 1 MONTH)) t;
-- ※ 관리자 화면 [로그관리 > 관리자 로그] (/sym/log/clg/SelectLoginLogList.do)와 동일한
--    LETTNADMINLOG 기준. 관리자별·메뉴별 상세 내역은 해당 화면에서 조회·캡처 가능
```

▸ **결과 캡처**

점검항목	당월건수	접속 관리자수	최초일시	최종일시	전월건수	증감률	판정
관리자 접속·행위	3,040	38	2026-07-01 00:08:56.000	2026-07-31 20:35:02.000	4,185	-27.4%	정상

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

2. 오류·장애 (WAS)

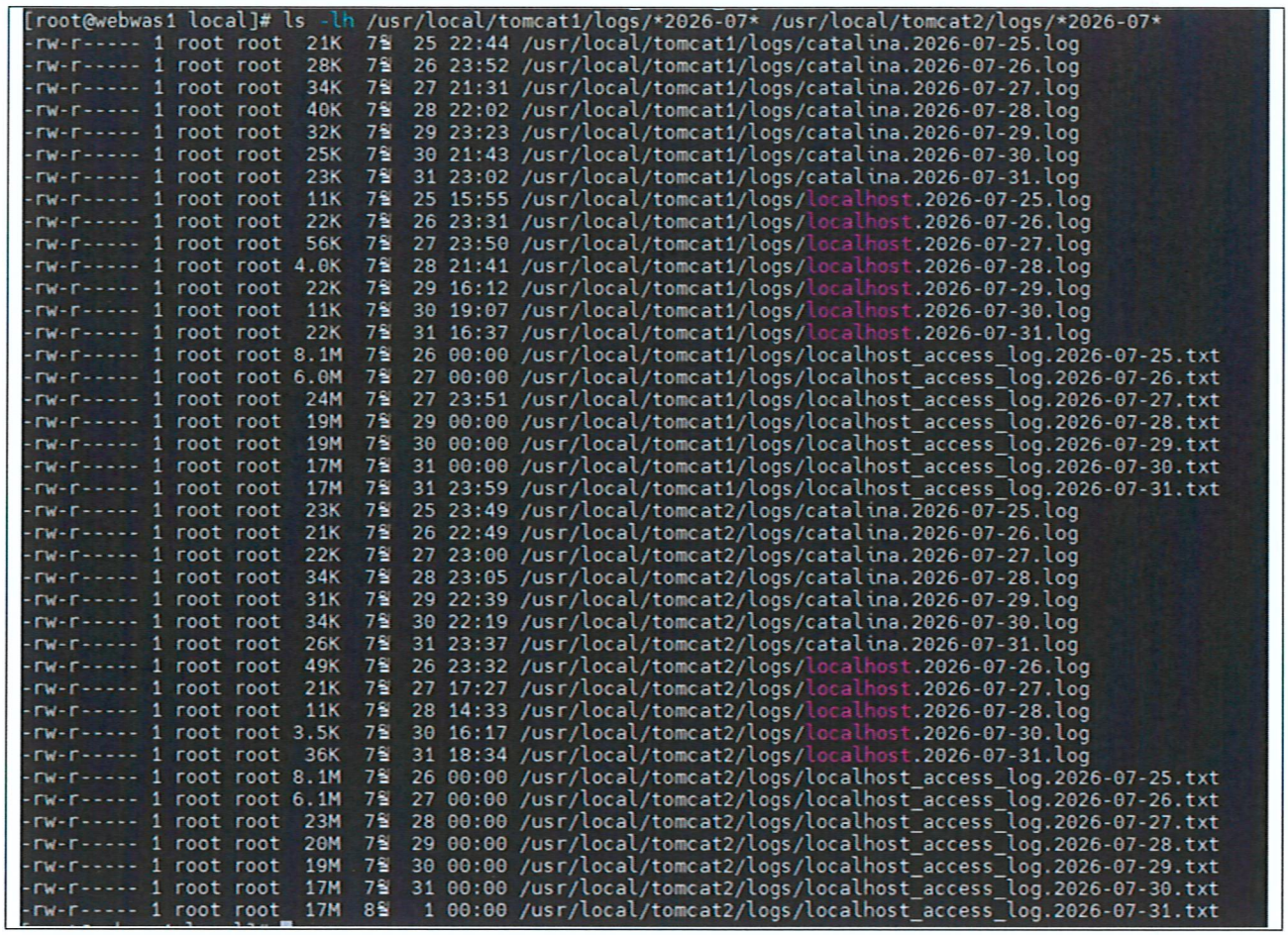
점검대상	tomcat1·tomcat2 표준출력(catalina.out) / log4j2	저장위치	파일	보관기간	1 년 이상

▸ **점검 명령어 (실행 후 결과 그리드를 그대로 캡처)**

```
# 점검 대상 월의 로그 목록(파일명·날짜·용량)만 출력 → 이 화면 캡처가 곧 증빙 (월만 바꿔서 실행)
ls -lh /usr/local/tomcat1/logs/*2026-08* /usr/local/tomcat2/logs/*2026-08*
# → catalina·localhost·localhost_access_log 일별 파일이 모두 잡힘. 날짜 빠진 일자 = 누락 의심


# (선택) 오류·장애 급증이 의심될 때만 추가 실행
grep -c "ERROR" /usr/local/tomcat1/logs/catalina.out /usr/local/tomcat2/logs/catalina.out
# ※ 운영 WAS 는 tomcat1·tomcat2 이중화 – 두 인스턴스 모두 포함
```

▸ **결과 캡처**



```
[root@webwas1 local]# ls -lh /usr/local/tomcat1/logs/*2026-07* /usr/local/tomcat2/logs/*2026-07*
-rw-r----- 1 root root  21K  7월  25 22:44 /usr/local/tomcat1/logs/catalina.2026-07-25.log
-rw-r----- 1 root root  28K  7월  26 23:52 /usr/local/tomcat1/logs/catalina.2026-07-26.log
-rw-r----- 1 root root  34K  7월  27 21:31 /usr/local/tomcat1/logs/catalina.2026-07-27.log
-rw-r----- 1 root root  40K  7월  28 22:02 /usr/local/tomcat1/logs/catalina.2026-07-28.log
-rw-r----- 1 root root  32K  7월  29 23:23 /usr/local/tomcat1/logs/catalina.2026-07-29.log
-rw-r----- 1 root root  25K  7월  30 21:43 /usr/local/tomcat1/logs/catalina.2026-07-30.log
-rw-r----- 1 root root  23K  7월  31 23:02 /usr/local/tomcat1/logs/catalina.2026-07-31.log
-rw-r----- 1 root root  11K  7월  25 15:55 /usr/local/tomcat1/logs/localhost.2026-07-25.log
-rw-r----- 1 root root  22K  7월  26 23:31 /usr/local/tomcat1/logs/localhost.2026-07-26.log
-rw-r----- 1 root root  56K  7월  27 23:50 /usr/local/tomcat1/logs/localhost.2026-07-27.log
-rw-r----- 1 root root 4.0K  7월  28 21:41 /usr/local/tomcat1/logs/localhost.2026-07-28.log
-rw-r----- 1 root root  22K  7월  29 16:12 /usr/local/tomcat1/logs/localhost.2026-07-29.log
-rw-r----- 1 root root  11K  7월  30 19:07 /usr/local/tomcat1/logs/localhost.2026-07-30.log
-rw-r----- 1 root root  22K  7월  31 16:37 /usr/local/tomcat1/logs/localhost.2026-07-31.log
-rw-r----- 1 root root 8.1M  7월  26 00:00 /usr/local/tomcat1/logs/localhost_access_log.2026-07-25.txt
-rw-r----- 1 root root 6.0M  7월  27 00:00 /usr/local/tomcat1/logs/localhost_access_log.2026-07-26.txt
-rw-r----- 1 root root  24M  7월  27 23:51 /usr/local/tomcat1/logs/localhost_access_log.2026-07-27.txt
-rw-r----- 1 root root  19M  7월  29 00:00 /usr/local/tomcat1/logs/localhost_access_log.2026-07-28.txt
-rw-r----- 1 root root  19M  7월  30 00:00 /usr/local/tomcat1/logs/localhost_access_log.2026-07-29.txt
-rw-r----- 1 root root  17M  7월  31 00:00 /usr/local/tomcat1/logs/localhost_access_log.2026-07-30.txt
-rw-r----- 1 root root  17M  7월  31 23:59 /usr/local/tomcat1/logs/localhost_access_log.2026-07-31.txt
-rw-r----- 1 root root  23K  7월  25 23:49 /usr/local/tomcat2/logs/catalina.2026-07-25.log
-rw-r----- 1 root root  21K  7월  26 22:49 /usr/local/tomcat2/logs/catalina.2026-07-26.log
-rw-r----- 1 root root  22K  7월  27 23:00 /usr/local/tomcat2/logs/catalina.2026-07-27.log
-rw-r----- 1 root root  34K  7월  28 23:05 /usr/local/tomcat2/logs/catalina.2026-07-28.log
-rw-r----- 1 root root  31K  7월  29 22:39 /usr/local/tomcat2/logs/catalina.2026-07-29.log
-rw-r----- 1 root root  34K  7월  30 22:19 /usr/local/tomcat2/logs/catalina.2026-07-30.log
-rw-r----- 1 root root  26K  7월  31 23:37 /usr/local/tomcat2/logs/catalina.2026-07-31.log
-rw-r----- 1 root root  49K  7월  26 23:32 /usr/local/tomcat2/logs/localhost.2026-07-26.log
-rw-r----- 1 root root  21K  7월  27 17:27 /usr/local/tomcat2/logs/localhost.2026-07-27.log
-rw-r----- 1 root root  11K  7월  28 14:33 /usr/local/tomcat2/logs/localhost.2026-07-28.log
-rw-r----- 1 root root 3.5K  7월  30 16:17 /usr/local/tomcat2/logs/localhost.2026-07-30.log
-rw-r----- 1 root root  36K  7월  31 18:34 /usr/local/tomcat2/logs/localhost.2026-07-31.log
-rw-r----- 1 root root 8.1M  7월  26 00:00 /usr/local/tomcat2/logs/localhost_access_log.2026-07-25.txt
-rw-r----- 1 root root 6.1M  7월  27 00:00 /usr/local/tomcat2/logs/localhost_access_log.2026-07-26.txt
-rw-r----- 1 root root  23M  7월  28 00:00 /usr/local/tomcat2/logs/localhost_access_log.2026-07-27.txt
-rw-r----- 1 root root  20M  7월  29 00:00 /usr/local/tomcat2/logs/localhost_access_log.2026-07-28.txt
-rw-r----- 1 root root  19M  7월  30 00:00 /usr/local/tomcat2/logs/localhost_access_log.2026-07-29.txt
-rw-r----- 1 root root  17M  7월  31 00:00 /usr/local/tomcat2/logs/localhost_access_log.2026-07-30.txt
-rw-r----- 1 root root  17M  8월   1 00:00 /usr/local/tomcat2/logs/localhost_access_log.2026-07-31.txt
```

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

3. 개인정보 처리 이력

점검대상	personal_information_log (개인정보 처리 이력 전용 · 2026-08 신설)	저장위치	DB	보관기간	1 년 이상 (해당 시 2 년)

▸ **점검 SQL (실행 후 결과 그리드를 그대로 캡처)**

```
SET @m := '2026-08-01';  -- ★ 점검 대상 월의 1일 (이 줄만 수정)
SELECT '개인정보 처리 이력' AS `점검항목`,
       COUNT(*)                  AS `당월건수`,
       SUM(ACTION = '조회')      AS `조회`,
       SUM(ACTION = '수정')      AS `수정`,
       SUM(ACTION = '삭제')      AS `삭제`,
       SUM(ACTION = '다운로드') AS `다운로드`,
       SUM(ACTION = '생성')      AS `생성`,
       COUNT(DISTINCT USER_ID)  AS `처리자수`,
       CASE WHEN COUNT(*) = 0 THEN '확인필요(0건)' ELSE '정상' END AS `판정`
  FROM personal_information_log
 WHERE FRST_REGIST_PNTTM >= @m
   AND FRST_REGIST_PNTTM <  DATE_ADD(@m, INTERVAL 1 MONTH);
-- ※ 개인정보 처리 이력 전용 테이블(2026-08 운영 반영, PMS #5858) — 조회·수정·삭제·다운로드 행위별 기록
--    컬럼: USER_ID(처리자)·ACCESS_IP·URI·MENU_NM(화면명)·ACTION(행위)·TARGET(대상: 회원ID 또는 검색조건·건수)
```

```
-- ※ 2026-07 분까지는 신설 전이므로 기존 테이블(LETTHEMPLYRINFOCHANGEDTLS·LETTNADMINLOG) 기준 점검 결과 첨부
```

▸ **결과 캡처**

점검항목	당월건수	조회	수정	삭제	다운로드	생성	처리자수	판정
개인정보 처리 이력	0	[NULL]	[NULL]	[NULL]	[NULL]	[NULL]	0	확인필요(0건)

※ 신규 생성 테이블이라 코드 운영 반영 후 데이터 저장됨

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

4. OS 접속 로그

점검대상	/var/log/secure, audit.log, messages, lastlog	**저장위치**	파일	**보관기간**	1 년 이상

▸ **점검 명령어 (실행 후 결과 그리드를 그대로 캡처)**

```
# OS 로그 파일 목록(파일명·날짜·용량)을 통째로 출력 → 이 화면 캡처가 곧 증빙
sudo ls -lh /var/log/secure* /var/log/messages* /var/log/audit/


# (선택) 비인가 접근이 의심될 때만 추가 실행
sudo grep -c "Failed password" /var/log/secure
```

▸ **결과 캡처**

```
[root@webwas1 local]# ls -lh /var/log/secure* /var/log/messages* /var/log/audit/
-rw-------  1 root root  41K  8월   5 12:30 /var/log/messages
-rw-------  1 root root  89K  7월  12 03:30 /var/log/messages-20260712
-rw-------  1 root root 177K  7월  19 03:10 /var/log/messages-20260719
-rw-------  1 root root  86K  7월  26 03:20 /var/log/messages-20260726
-rw-------  1 root root  86K  8월   2 03:40 /var/log/messages-20260802
-rw-------  1 root root 424K  8월   5 12:30 /var/log/secure
-rw-------  1 root root 876K  7월  12 03:30 /var/log/secure-20260712
-rw-------  1 root root 899K  7월  19 03:10 /var/log/secure-20260719
-rw-------  1 root root 883K  7월  26 03:20 /var/log/secure-20260726
-rw-------  1 root root 888K  8월   2 03:40 /var/log/secure-20260802

/var/log/audit/:
합계 40M
-rw------- 1 root root 7.5M  8월   5 12:30 audit.log
-r-------- 1 root root 8.1M  7월   8 14:40 audit.log.1
-r-------- 1 root root 8.1M  6월   7 21:40 audit.log.2
-r-------- 1 root root 8.1M  5월   8 15:00 audit.log.3
-r-------- 1 root root 8.1M  4월  11 22:50 audit.log.4
[root@webwas1 local]#
```

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

5. DB 서버 로그

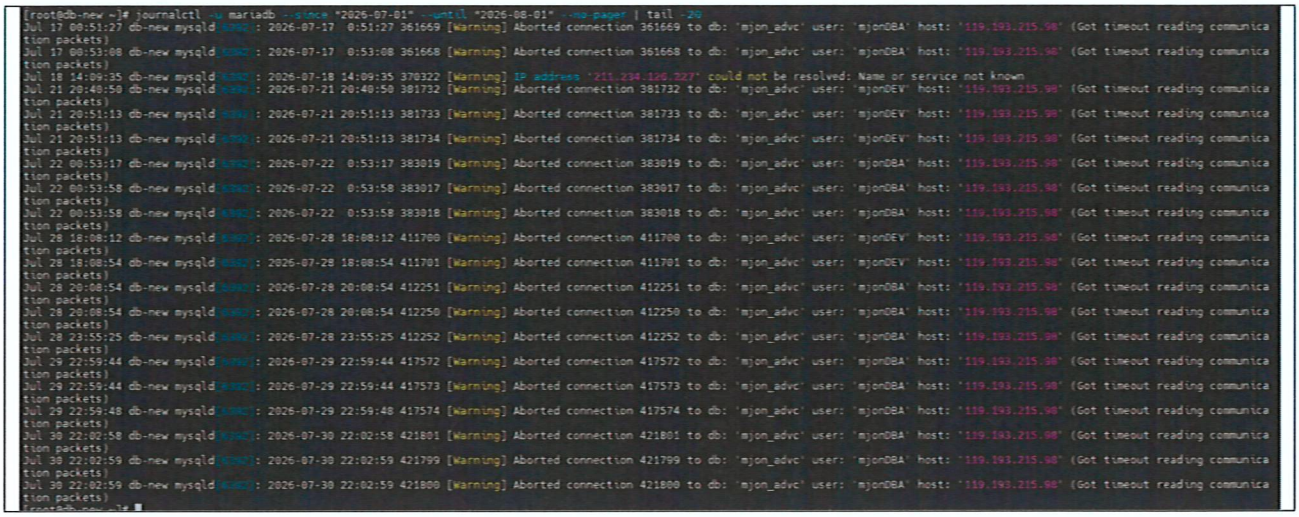
점검대상	MariaDB 오류 로그 — 현재 journald 기록 (db-new 서버)	저장위치	파일/DB	보관기간	1 년 이상

▸ **점검 명령어 (실행 후 결과 그리드를 그대로 캡처)**

```
# DB 서버(db-new)에서 당월 오류·경고 기록 확인 → 이 출력 캡처 (월만 바꿔서 실행)
journalctl -u mariadb --since "2026-08-01" --until "2026-09-01" --no-pager | tail -20


# ※ 가이드라인 4.2 요건은 DB '오류·장애' 로그 – 현재 log_error 미설정으로 journald 에 기록됨
# ※ log_error 파일 지정(/var/log/mariadb/) 후에는 ls -lh /var/log/mariadb/ 캡처로 대체
```

▸ **결과 캡처**



판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

6. API 서버 로그

점검대상	/data/tomcat/tomcat_api_9100_2023_0711/logs (logback 일별 롤링)	저장위치	파일	보관기간	1 년 이상

▸ **점검 명령어 (실행 후 결과 그리드를 그대로 캡처)**

```
# 당월 로그 파일 목록(일별 파일명·날짜·용량)을 통째로 출력 → 이 화면 캡처가 곧 증빙
ls -lh /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-08-*.log
# ※ logback-YYYY-MM-DD.0.log 일별 롤링 – 목록에서 날짜가 빠진 일자 = 누락 의심 (월만 바꿔서 실행)
```

▸ **결과 캡처**

```
[root@was logs]# ls -lh /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-*.log
-rw-r--r-- 1 root root 2.6M Jul  1 20:54 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-01.0.log
-rw-r--r-- 1 root root 4.3M Jul  2 23:12 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-02.0.log
-rw-r--r-- 1 root root 3.1M Jul  3 23:59 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-03.0.log
-rw-r--r-- 1 root root 9.9M Jul  4 20:21 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-04.0.log
-rw-r--r-- 1 root root 1.3M Jul  5 23:37 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-05.0.log
-rw-r--r-- 1 root root 8.6M Jul  6 22:52 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-06.0.log
-rw-r--r-- 1 root root 7.5M Jul  7 21:56 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-07.0.log
-rw-r--r-- 1 root root 7.4M Jul  8 23:13 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-08.0.log
-rw-r--r-- 1 root root 6.3M Jul  9 23:33 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-09.0.log
-rw-r--r-- 1 root root 5.5M Jul 10 23:07 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-10.0.log
-rw-r--r-- 1 root root 3.2M Jul 11 23:32 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-11.0.log
-rw-r--r-- 1 root root 537K Jul 12 23:15 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-12.0.log
-rw-r--r-- 1 root root 7.7M Jul 13 23:56 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-13.0.log
-rw-r--r-- 1 root root 7.1M Jul 14 22:55 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-14.0.log
-rw-r--r-- 1 root root 5.0M Jul 15 22:38 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-15.0.log
-rw-r--r-- 1 root root 5.2M Jul 16 23:59 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-16.0.log
-rw-r--r-- 1 root root 749K Jul 17 23:23 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-17.0.log
-rw-r--r-- 1 root root 1.7M Jul 18 23:45 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-18.0.log
-rw-r--r-- 1 root root 6.0M Jul 19 22:38 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-19.0.log
-rw-r--r-- 1 root root  11M Jul 20 23:55 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-20.0.log
-rw-r--r-- 1 root root 5.8M Jul 21 23:56 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-21.0.log
-rw-r--r-- 1 root root  12M Jul 22 23:03 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-22.0.log
-rw-r--r-- 1 root root 5.2M Jul 23 23:31 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-23.0.log
-rw-r--r-- 1 root root 6.7M Jul 24 22:19 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-24.0.log
-rw-r--r-- 1 root root 1.4M Jul 25 23:26 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-25.0.log
-rw-r--r-- 1 root root 2.9M Jul 26 23:57 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-26.0.log
-rw-r--r-- 1 root root 5.8M Jul 27 22:51 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-27.0.log
-rw-r--r-- 1 root root 3.4M Jul 28 20:59 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-28.0.log
-rw-r--r-- 1 root root 4.8M Jul 29 22:56 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-29.0.log
-rw-r--r-- 1 root root 5.3M Jul 30 23:31 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-30.0.log
-rw-r--r-- 1 root root 3.7M Jul 31 23:40 /data/tomcat/tomcat_api_9100_2023_0711/logs/logback-2026-07-31.0.log
```

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

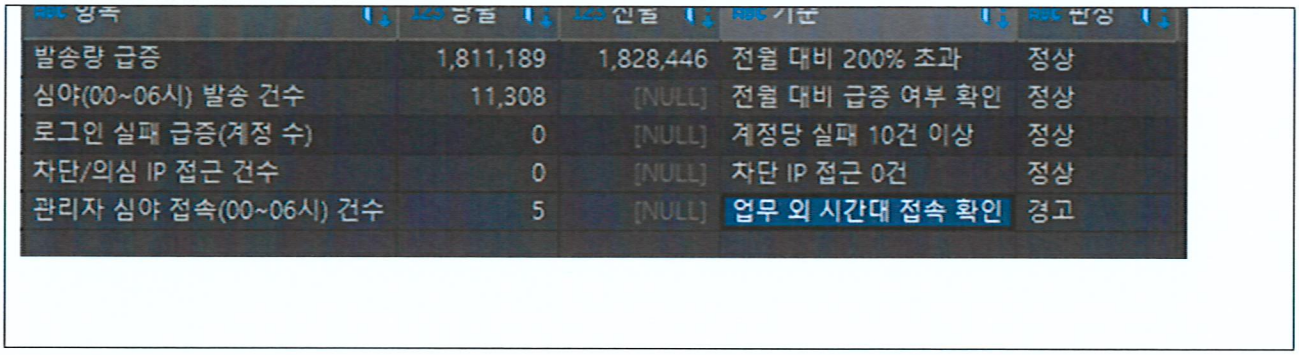
③ 이상징후 점검

아래 쿼리 하나를 실행하면 이상징후 자동 점검 항목별 결과·판정이 표로 나온다. 결과 그리드를 그대로 캡처하여 첨부하고, 경고·이상 행은 ④ 후속조치 기록에 기재한다.

```
SET @m := '2026-08-01';  -- ★ 점검 대상 월의 1일 (이 줄만 수정)
SELECT '발송량 급증' AS `항목`, t.cur AS `당월`, t.prev AS `전월`,
       '전월 대비 200% 초과' AS `기준`,
       CASE WHEN t.prev > 0 AND t.cur > t.prev*2 THEN '경고' ELSE '정상' END AS `판정`
  FROM (SELECT SUM(REQ_DATE >= @m) AS cur, SUM(REQ_DATE < @m) AS prev
          FROM MJ_MSG_DATA
         WHERE REQ_DATE >= DATE_SUB(@m, INTERVAL 1 MONTH)
           AND REQ_DATE <  DATE_ADD(@m, INTERVAL 1 MONTH)) t
UNION ALL
SELECT '심야(00~06시) 발송 건수', COUNT(*), NULL, '전월 대비 급증 여부 확인',
       CASE WHEN COUNT(*) > 30000 THEN '경고' ELSE '정상' END
  FROM MJ_MSG_DATA
 WHERE REQ_DATE >= @m AND REQ_DATE < DATE_ADD(@m, INTERVAL 1 MONTH)
   AND HOUR(REQ_DATE) BETWEEN 0 AND 5
UNION ALL
SELECT '로그인 실패 급증(계정 수)', COUNT(*), NULL, '계정당 실패 10건 이상',
       CASE WHEN COUNT(*) > 0 THEN '경고' ELSE '정상' END
  FROM (SELECT LOGIN_ID FROM MJ_LOGIN_LOG
         WHERE LOGIN_YN = 'N'
           AND LOGIN_DT >= @m AND LOGIN_DT < DATE_ADD(@m, INTERVAL 1 MONTH)
         GROUP BY LOGIN_ID HAVING COUNT(*) >= 10) x
UNION ALL
SELECT '차단/의심 IP 접근 건수', COUNT(*), NULL, '차단 IP 접근 0건',
       CASE WHEN COUNT(*) > 0 THEN '이상' ELSE '정상' END
  FROM MJ_LOGIN_LOG L JOIN HACK_IP H ON L.LOGIN_IP = H.ACCESS_IP
 WHERE H.BLOCK_YN = 'Y'
   AND L.LOGIN_DT >= @m AND L.LOGIN_DT < DATE_ADD(@m, INTERVAL 1 MONTH)
UNION ALL
SELECT '관리자 심야 접속(00~06시) 건수', COUNT(*), NULL, '업무 외 시간대 접속 확인',
       CASE WHEN COUNT(*) > 0 THEN '경고' ELSE '정상' END
  FROM LETTNADMINLOG a
 WHERE a.CONECT_ID != ''
   AND EXISTS (SELECT 1 FROM lettnemplyrinfo c WHERE c.ESNTL_ID = a.CONECT_ID)  -- 관리자만
   AND a.CREAT_DT >= @m AND a.CREAT_DT < DATE_ADD(@m, INTERVAL 1 MONTH)
   AND HOUR(a.CREAT_DT) BETWEEN 0 AND 5
UNION ALL
SELECT '발송 실패 건수', SUM(REQ_DATE >= @m AND SWF = 'F'), SUM(REQ_DATE < @m AND SWF = 'F'),
       '전월 대비 급증 여부 확인',
       CASE WHEN SUM(REQ_DATE < @m AND SWF = 'F') > 0
            AND SUM(REQ_DATE >= @m AND SWF = 'F') > SUM(REQ_DATE < @m AND SWF = 'F') * 2
           THEN '경고' ELSE '정상' END
  FROM MJ_MSG_DATA
 WHERE REQ_DATE >= DATE_SUB(@m, INTERVAL 1 MONTH)
   AND REQ_DATE <  DATE_ADD(@m, INTERVAL 1 MONTH)
UNION ALL
SELECT '개인정보 다운로드 건수', COUNT(*), NULL, '대량(50건 초과) 여부 확인',
       CASE WHEN COUNT(*) > 50 THEN '경고' ELSE '정상' END
  FROM personal_information_log
```

```
 WHERE ACTION = '다운로드'
   AND FRST_REGIST_PNTTM >= @m AND FRST_REGIST_PNTTM < DATE_ADD(@m, INTERVAL 1
MONTH);
-- ※ SWF: 발송결과(S=성공/F=실패) – 값 체계는 운영 데이터로 확인 후 조정
-- ※ 임계치(심야 30,000건·실패 10건·다운로드 50건 등)는 운영 기준으로 조정, 조정 시 사유 기재
```

▸ **결과 캡처**



항목	당월	전월	기준	판정
발송량 급증	1,811,189	1,828,446	전월 대비 200% 초과	정상
심야(00~06시) 발송 건수	11,308	[NULL]	전월 대비 급증 여부 확인	정상
로그인 실패 급증(계정 수)	0	[NULL]	계정당 실패 10건 이상	정상
차단/의심 IP 접근 건수	0	[NULL]	차단 IP 접근 0건	정상
관리자 심야 접속(00~06시) 건수	5	[NULL]	업무 외 시간대 접속 확인	경고

판정 확인	정상 ☑ 경고 ☐ 이상 ☐ (쿼리 판정과 달리 판단 시 사유:)
조치·비고	

④ 이상징후 후속조치 기록

발견일시	이상 내용	원인	조치 내용	보고 (담당→책임자)	조치기한	종결확인 (일자/확인자)

◆ **판정 범례 : 정상 = 이상 없음 / 경고 = 관찰 필요 / 이상 = 조치 필요**

◆ **첨부(필수) : 각 항목의 쿼리(명령) 실행 결과 그리드 캡처 — 건수·기간·판정(또는 파일명·사이즈)이 보이도록.**

◆ **보관 : 본 보고서는 Redmine 월별 이슈에 첨부하여 1 년 이상 보관하며, 권한자만 접근·수정한다.**